Userview Theme Plugin

- Usages
- Abstract Class
    - org.joget.apps.userview.model.UserviewV5Theme
        - Method Detail
            - Overridable Methods
                - handleContentError
                - handlePageNotFound
                - getLayout
                - getHeader
                - getFooter
                - getContentContainer
                - getMenus
                - getJsCssLib
                - getCss
                - getJs
                - getMetas
                - getHead
                - getFavIconLink
                - getLoginForm
                - decorateCategoryLabel
            - Utility Methods
                - getUserview
                - setUserview
                - getRequestParameters
                - setRequestParameters
                - getRequestParameter
                - getRequestParameterString
            - Deprecated Methods
                - getCss
                - getJavascript
                - public java.lang.String getHeader()getHeader
                - getFooter
                - getPageTop
                - getPageBottom
                - getBeforeContent
    - org.joget.apps.userview.model.UserviewTheme
        - Method Detail
            - Abstract Methods
                - getCss
                - getJavascript
                - getHeader
                - getFooter
                - getPageTop
                - getPageBottom
                - getBeforeContent
            - Utility Methods
                - getUserview
                - setUserview
                - getRequestParameters
                - setRequestParameters
                - getRequestParameter
                - getRequestParameterString
- V5 Predefined Theme Template
    - Predefined Variables
    - Templates
        - /templates/userview/pageNotFound.ftl
        - /templates/userview/layout.ftl
        - /templates/userview/header.ftl
        - /templates/userview/footer.ftl
        - /templates/userview/contentContainer.ftl
        - /templates/userview/menus.ftl
        - /templates/userview/head.ftl
        - /templates/userview/login.ftl
    - How to use and override V5 template
        - Reuse the predefined template
        - Override or replace the predefined template
- Plugin Properties Options
- Tutorials
- Related Community Plugins

Usages

- Used for providing a different look and feel for a Userview. Please refer to UI Theme.
- A useable Userview Theme plugin must extends org.joget.apps.userview.model.UserviewV5Theme or org.joget.apps.userview.model.UserviewTheme abstract class.
- Since version 5.0, It is encourage to use org.joget.apps.userview.model.UserviewV5Theme instead of org.joget.apps.userview.model.UserviewTheme abstract class for Userview Theme plugin
- development as the new theme provided more flexibility to control how you want your theme to display to user.

Abstract Class

org.joget.apps.userview.model.UserviewV5Theme

- Under wflow-core module
- Extended org.joget.plugin.base.ExtDefaultPlugin. Please refer to Plugin Base Abstract Class and Interface.
- Implemented org.joget.plugin.property.model.PropertyEditable. Please refer to Plugin Base Abstract Class and Interface.
- A base abstract class to develop a Userview Theme plugin for version v5.0 onward.
- Please refer to V5 Predefined Theme Template on template options.
- Only need to override the function accordingly to change part of the display.

Method Detail

Overridable Methods

handleContentError

```
public java.lang.String handleContentError(java.lang.Exception e, java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template to handle error when retrieving userview content.

handlePageNotFound

```
public java.lang.String handlePageNotFound(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template to handle page not found. Default using /templates/userview/pageNotFound.ftl as template.

getLayout

```
public java.lang.String getLayout(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template to handle theme layout. Default using /templates/userview/layout.ftl as template.

getHeader

```
public java.lang.String getHeader(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template to handle page header. Default using /templates/userview/header.ftl as template.

getFooter

```
public java.lang.String getFooter(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template to handle page footer. Default using /templates/userview/footer.ftl as template.

getContentContainer

```
public java.lang.String getContentContainer(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template to handle userview menu content. Default using /templates/userview/contentContainer.ftl as template.

getMenus

```
public java.lang.String getMenus(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template to handle menus. Default using /templates/userview/menus.ftl as template.

getJsCssLib

```
public java.lang.String getJsCssLib(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template for putting javascript and css link for getHead() template.

getCss

```
public java.lang.String getCss(java.util.Map<java.lang.String, java.lang.Object> data)
```

Gets dynamic generated CSS for getHead() template

getJs

```
public java.lang.String getJs(java.util.Map<java.lang.String, java.lang.Object> data)
```

Gets dynamic generated javascript for getHead() template

getMetas

```
public java.lang.String getMetas(java.util.Map<java.lang.String, java.lang.Object> data)
```

Gets dynamic generated meta data for getHead() template

getHead

```
public java.lang.String getHead(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template to handle for <head> tag. Default using /templates/userview/head.ftl as template.

getFavIconLink

```
public java.lang.String getFavIconLink(java.util.Map<java.lang.String, java.lang.Object> data)
```

Gets the fav icon relative path for getHead() template.

getLoginForm

```
public java.lang.String getLoginForm(java.util.Map<java.lang.String, java.lang.Object> data)
```

HTML template for login form. Default using /templates/userview/login.ftl as template.

decorateCategoryLabel

```
public java.lang.String decorateCategoryLabel(org.joget.apps.userview.model.UserviewCategory category)
```

HTML template for menu category label.

Utility Methods

getUserview

```
public org.joget.apps.userview.model.Userview getUserview()
```

Gets userview which using this theme

setUserview

```
public void setUserview(org.joget.apps.userview.model.Userview userview)
```

Sets userview which using this theme

getRequestParameters

```
public java.util.Map getRequestParameters()
```

Gets request parameters

setRequestParameters

```
public void setRequestParameters(java.util.Map requestParameters)
```

Sets request parameters

getRequestParameter

```
public java.lang.Object getRequestParameter(java.lang.String requestParameter)
```

Convenience method to get a parameter value

getRequestParameterString

```
public java.lang.String getRequestParameterString(java.lang.String requestParameter)
```

Convenience method to get a parameter String value.

Return Empty string instead of NULL.

Deprecated Methods

getCss

```
public java.lang.String getCss()
```

Deprecated not use for UserviewV5Theme

getJavascript

```
public java.lang.String getJavascript()
```

Deprecated not use for UserviewV5Theme

public java.lang.String getHeader()getHeader

Deprecated not use for UserviewV5Theme

getFooter

```
public java.lang.String getFooter()
```

Deprecated not use for UserviewV5Theme

getPageTop

```
public java.lang.String getPageTop()
```

Deprecated not use for UserviewV5Theme

getPageBottom

```
public java.lang.String getPageBottom()
```

Deprecated not use for UserviewV5Theme

getBeforeContent

```
public java.lang.String getBeforeContent()
```

Deprecated not use for UserviewV5Theme

org.joget.apps.userview.model.UserviewTheme

- Under wflow-core module
- Extended org.joget.plugin.base.ExtDefaultPlugin. Please refer to Plugin Base Abstract Class and Interface.
- Implemented org.joget.plugin.property.model.PropertyEditable. Please refer to Plugin Base Abstract Class and Interface.
- A base abstract class to develop a Userview Theme plugin.

Method Detail

Abstract Methods

getCss

```
public abstract java.lang.String getCss()
```

Return css to inject in <head> tag

getJavascript

```
public abstract java.lang.String getJavascript()
```

Return javascript to inject in <head> tag

getHeader

```
public abstract java.lang.String getHeader()
```

Return HTML template to replace default header

getFooter

```
public abstract java.lang.String getFooter()
```

Return HTML template to replace default footer

getPageTop

```
public abstract java.lang.String getPageTop()
```

Return HTML template to inject before the page container

getPageBottom

```
public abstract java.lang.String getPageBottom()
```

Return HTML template to inject after the page container

getBeforeContent

```
public abstract java.lang.String getBeforeContent()
```

Return HTML template to inject before content

Utility Methods

getUserview

```
public org.joget.apps.userview.model.Userview getUserview()
```

Gets userview which using this theme

setUserview

```
public void setUserview(org.joget.apps.userview.model.Userview userview)
```

Sets userview which using this theme

getRequestParameters

```
public java.util.Map getRequestParameters()
```

Gets request parameters

setRequestParameters

```
public void setRequestParameters(java.util.Map requestParameters)
```

Sets request parameters

getRequestParameter

```
public java.lang.Object getRequestParameter(java.lang.String requestParameter)
```

Convenience method to get a parameter value

getRequestParameterString

```
public java.lang.String getRequestParameterString(java.lang.String requestParameter)
```

Convenience method to get a parameter String value.

Return Empty string instead of NULL.

V5 Predefined Theme Template

Predefined Variables

- Predefined variable with value for template usage template.

Name	Template	Data Type	Default Value	Description
base_link	-	java.lang.String		Base URL of current Userview
body_classes	layout.ftl	java.lang.String		Used by system to add the following CSS classes and current locale code as CSS class to <body> tag. embeded - When userview is in embed mode. rtl - When System setting set display from right to left or language set to Arabic.
body_id	layout.ftl	java.lang.String		Use current userview menu page Id or Custom Id as <body> tag id
build_number	-	java.lang.String		Joget build number
categories_container_id	menus.ftl	java.lang.String	category-container	Id of the container which contains all menu categories
categories_container_classes	menus.ftl	java.lang.String	nav nav-tabs nav-stacked main-menu	CSS classes of the container which contains all menu categories
category_classes	menus.ftl	java.lang.String	category	CSS classes of menu category
combine_single_menu_category	menus.ftl	boolean	false	Flag to display the only menu in a category to replace the category as 1st level menu.
content	contentContainer.ftl	java.lang.String		Main content of the current page. Content from the Userview Menu or login form.
content_container	layout.ftl	java.lang.String		Value from theme getContentContainer method
content_id	contentContainer.ftl	java.lang.String	content	Id of content container
context_path	login.ftl	java.lang.String		Context Path of the Joget Server URL
css	head.ftl	java.lang.String		Value from theme getCss method
current_category_classes	menus.ftl	java.lang.String	current-category active	Extra CSS classes for current menu category
current_menu_classes	menus.ftl	java.lang.String	current active	Extra CSS classes for current menu
embed	contentContainer.ftl	boolean		Flag to decide whether userview is needed to display in embed mode
fav_icon_link	head.ftl	java.lang.String		Value from theme getFavIconLink method
first_category_classes	menus.ftl	java.lang.String	first	Extra CSS classes for first menu category
first_menu_classes	menus.ftl	java.lang.String	first	Extra CSS classes for first menu in the menu category
footer	layout.ftl	java.lang.String		Value from theme getFooter method
head	layout.ftl	java.lang.String		Value from theme getHead method
header	layout.ftl	java.lang.String		Value from theme getHeader method
hide_nav	contentContainer.ftl	boolean	false	Flag to hide navigator
home_page_link	pageNotFound.ftl, header.ftl	java.lang.String		URL to the userview home page menu
is_logged_in	header.ftl	boolean		Flag to decide whether there is a logged in user
is_login_page	-	boolean		Whether or not the current page is a login form page
joget_header	head.ftl	java.lang.String		Additional HTML template injected by system to the header
joget_footer	layout.ftl	java.lang.String		Additional HTML template injected by system to the footer
js	head.ftl	java.lang.String		Value from theme getJs method
js_css_lib	head.ftl	java.lang.String		Value from theme getJsCssLib method
last_category_classes	menus.ftl	java.lang.String	last	Extra CSS classes for last menu category
last_menu_classes	menus.ftl	java.lang.String	last	Extra CSS classes for last menu in the menu category
locale	layout.ftl	java.lang.String		Current language code of the system
login_error_classes	login.ftl	java.lang.String	form-errors alert alert-warning	CSS classes for login error message container

login_exception	login.ftl	java.lang.String		To keep exception message when a login attempt failure.
login_form_after	login.ftl	java.lang.String		Value from "Custom HTML (After Login Form)" under Userview Builder - Settings
login_form_before	login.ftl	java.lang.String		Value from "Custom HTML (Before Login Form)" under Userview Builder - Settings
login_form_footer	login.ftl	java.lang.String		HTML template retrieved from DirectoryUtil.getLoginFormFooter(). Used by User Security Implementation.
login_link	header.ftl	java.lang.String		URL to the login form
logout_link	header.ftl	java.lang.String		URL to log out from userview
main_container_id	contentContainer.ftl	java.lang.String	main	CSS classes for the main container
menus	contentContainer.ftl			Value from theme getMenus method
menu_classes	menus.ftl	java.lang.String	menu	CSS classes for the menu
menus_container_classes	menus.ftl	java.lang.String	menu-container	CSS classes for the container which contains menus
metas	head.ftl	java.lang.String		Value from theme getMetas method
nav_id	menus.ftl	java.lang.String	navigation	Id of the navigator container
nav_classes	menus.ftl	java.lang.String	nav-collapse sidebar-nav	CSS classes of the navigator container
params	-	java.util.Map<java.lang.String, java.lang.Object>		Received Request Parameters
title	head.ftl	java.lang.String		Userview name and the current menu label
right_to_left	-	boolean		Flag to decide whether or not userview need to display from right to left
sidebar_id	contentContainer.ftl	java.lang.String	sidebar	Id of the sidebar container
user	-	org.joget.directory.model.User		Current logged in User object
username	-	java.lang.String		Current logged in username
userview	header.ftl, footer.ftl, menus.ftl, login.ftl	org.joget.apps.userview.model.Userview		Userview object which using this theme

Templates

- All these template are used by org.joget.apps.userview.model.UserviewV5Theme abstract class.
- Override the abstract class method to set different value for the variables used in template.
- All templates are constructed using FreeMaker syntax.

/templates/userview/pageNotFound.ftl

```
<h3>@@ubuilder.pageNotFound@@</h3>
@@ubuilder.pageNotFound.message@@
<br><br>
@@ubuilder.pageNotFound.explanation@@
<p> </p>
<p> </p>
<p>
    <a href="${home_page_link!}">@@ubuilder.pageNotFound.backToMain@@</a>
</p>
```

/templates/userview/layout.ftl

```
<!DOCTYPE html>
<html lang="${locale!}">
    ${html_inner_before!}
    <head>
        ${head_inner_before!}
        ${head!}
        ${head_inner_after!}
    </head>
    <body id="${body_id!}" class="${body_classes!}">
        ${body_inner_before!}
        ${page_before!}
        <div id="page">
            ${page_inner_before!}
            ${header!}
            ${content_container!}
            ${footer!}
            ${page_inner_after!}
        </div>
        ${page_after!}
        ${joget_footer!}
        ${body_inner_after!}
    </body>
    ${html_inner_after!}
</html>
```

/templates/userview/header.ftl

```
${header_before!}
<header class="${header_classes!}">
    ${header_inner_before!}
    ${header_info_before!}
    <div id="header-info" class="${header_info_classes!}">
        ${header_info_inner_before!}
        ${header_name_before!}
        <div id="header-name" class="${header_name_classes!}">
            ${header_name_inner_before!}
            <a href="${home_page_link!}" id="header-link" class="${header_link_classes!}">
                <span>${userview.properties.name!}</span>
            </a>
            ${header_name_inner_after!}
        </div>
        ${header_name_after!}
        ${header_description_before!}
        <div id="header-description" class="${header_description_classes!}">
            ${header_description_inner_before!}
            <span id="description" class="${header_description_span_classes!}">${userview.properties.
description!}</span>
            ${header_description_inner_after!}
        </div>
        ${header_description_after!}
        <div class="clearfix"></div>
        ${header_info_inner_after!}
    </div>
    ${header_info_after!}
    ${header_message_before!}
    <div id="header-message" class="${header_message_classes!}">
        ${header_message_inner_before!}
        <div id="header-welcome-message" class="${header_welcome_classes!}">
            <span id="welcomeMessage">${userview.properties.welcomeMessage!}</span>
        </div>
        <div id="header-logout-text" class="${header_logout_classes!}">
            <#if is_logged_in>
                <a href="${logout_link!}">
                    <span id="logoutText">${userview.properties.logoutText!}</span>
                </a>
            <#else>
                <a href="${login_link!}">
                    <span id="loginText">@@ubuilder.login@@</span>
                </a>
            </#if>
        </div>
        <div class="clearfix"></div>
        ${header_message_inner_after!}
    </div>
    ${header_message_after!}
    ${header_inner_after!}
</header>
${header_after!}
```

/templates/userview/footer.ftl

```
<div class="clearfix"></div>
${footer_before!}
<footer class="${footer_classes!}">
    ${footer_inner_before!}
    <div id="footer-message">
        <p><span id="footerMessage">${userview.properties.footerMessage!}</span></p>
    </div>
    ${footer_inner_after!}
</footer>
${footer_after!}
```

/templates/userview/contentContainer.ftl

```
${main_container_before!}
<div id="${main_container_id!}" class="${main_container_classes!}">
    <div class="${main_container_inner_classes!}">
        ${main_container_inner_before!}
        <#if !embed && !hide_nav>
            ${sidebar_before!}
            <div id="${sidebar_id!}" class="${sidebar_classes!}">
                ${sidebar_inner_before!}
                ${menus!}
                ${sidebar_inner_after!}
            </div>
            ${sidebar_after!}
        </#if>
        ${content_before!}
        <div id="${content_id!}" class="${content_classes!}">
            <main>
                ${content_inner_before!}
                ${content!}
                ${content_inner_after!}
            </main>
        </div>
        ${content_after!}
        ${main_container_inner_before!}
    </div>
</div>
${main_container_before!}
```

/templates/userview/menus.ftl

```
${nav_before!}
<nav id="${nav_id!}" class="${nav_classes!}">
    ${nav_inner_before!}
    ${categories_container_before!}
    <ul id="${categories_container_id!}" class="${categories_container_classes!}">
        ${categories_container_inner_before!}
        <#list userview.categories as category>
            <#if category.properties.hide! != 'yes' && category.menus?size gt 0>
                ${category_before!}
                <#assign cClass = category_classes!>
                <#if category_index == 0>
                    <#assign cClass = cClass + " " + first_category_classes!>
                </#if>
                <#if (category_index + 1) == userview.categories?size>
                    <#assign cClass = cClass + " " + last_category_classes!>
                </#if>
                <#if userview.currentCategory?? && category.properties.id == userview.currentCategory.
properties.id>
                    <#assign cClass = cClass + " " + current_category_classes!>
                </#if>
                <#assign firstMenu = category.menus[0]>
                <#if combine_single_menu_category! && category.menus?size == 1>
                    <li id="${firstMenu.properties.id}" class="${cClass}">
                        ${category_inner_before!}
                        ${firstMenu.menu}
                        ${category_inner_after!}
                    </li>
                <#else>
                    <li class="${cClass}">
                        ${category_inner_before!}
                        <a class="${category_label_classes!} dropdown" href="#"><span>${theme.
decorateCategoryLabel(category)}</span></a>
                        ${menu_container_before!}
                        <ul class="${menus_container_classes!}" >
                            ${menu_container_inner_before!}
                            <#list category.menus as menu>
                                <#assign mClass = menu_classes!>
                                <#if menu_index == 0>
                                    <#assign mClass = mClass + " " + first_menu_classes!>
                                </#if>
                                <#if (menu_index + 1) == category.menus?size>
                                    <#assign mClass = mClass + " " + last_menu_classes!>
                                </#if>
                                <#if userview.current?? && menu.properties.id == userview.current.properties.id>
                                    <#assign mClass = mClass + " " + current_menu_classes!>
                                </#if>
                                ${menu_before!}
                                <li id="${menu.properties.id!}" class="${mClass}">
                                    ${menu_inner_before!}
                                    ${menu.menu}
                                    ${menu_inner_after!}
                                </li>
                                ${menu_after!}
                            </#list>
                            ${menu_container_inner_after!}
                        </ul>
                        ${menu_container_after!}
                        ${category_inner_after!}
                    </li>
                </#if>
                ${category_after!}
            </#if>
        </#list>
        ${categories_container_inner_after!}
    </ul>
    ${categories_container_after!}
    ${nav_inner_after!}
</nav>
${nav_after!}
```

/templates/userview/head.ftl

```
${metas!}
<title>${title!}</title>
${joget_header!}
${js_css_lib!}
<script type="text/javascript">
    ${js!}
</script>
<style type="text/css">
    ${css!}
</style>
<link rel="shortcut icon" href="${fav_icon_link!}"/>
```

/templates/userview/login.ftl

```
<#if userview.params.login_error?? >
    ${login_error_before!}
    <div id="main-body-message" class="${login_error_classes!}">
        ${login_error_inner_before!}
        ${login_exception!}
        ${login_error_inner_after!}
    </div>
    ${login_error_after!}
</#if>
${login_form_before!}
<form id="loginForm" name="loginForm" action="${context_path!}/j_spring_security_check" method="POST">
    ${login_form_inner_before!}
    <table align="center">
        <tr><td><label>@@ubuilder.login.username@@: </label></td><td><input type='text' id='j_username'
name='j_username'/></td></tr>
        <tr><td><label>@@ubuilder.login.password@@: </label></td><td><input type='password' id='j_password'
name='j_password'/></td></tr>
        <tr><td> </td><td><input name="submit" class="form-button" type="submit" value="@@ubuilder.
login@@" /></td></tr>
        <tr><td colspan="2">
            ${login_form_footer!}
        </td></tr>
    </table>
    ${login_form_inner_after!}
</form>
${login_form_after!}
```

How to use and override V5 template

Reuse the predefined template

- To reuse the predefined template, just override the theme method and set your custom HTML to the available variable of the template.
- Example: To reuse the header.ftl template in getHeader method.

```
    @Override
    public String getHeader(Map<String, Object> data) {
        data.put("header_classes", "navbar");
        data.put("header_inner_before", "<div class=\"navbar-inner\"><div class=\"container-fluid\">");
        data.put("header_inner_after", "</div></div>");
        data.put("header_info_after", "<a id=\"mobile_menu\" class=\"btn btn-navbar\"><span class=\"icon-bar\"><
/span><span class=\"icon-bar\"></span><span class=\"icon-bar\"></span></a>");
        data.put("header_link_classes", "brand");
        data.put("header_info_classes", "inline-block");
        data.put("header_name_classes", "inline-block");
        data.put("header_description_classes", "inline-block visible-desktop");
        data.put("header_description_span_classes", "brand");
        return super.getHeader(data);
    }
```

Override or replace the predefined template

- To override or replace the predefined template, you just need to create your own template using FreeMaker syntax.
- Example: Using a custom template file in getHeader method.

```
    @Override
    public String getHeader(Map<String, Object> data) {
        return UserviewUtil.getTemplate(this, data, "/templates/userview/customHeader.ftl");
    }
```

Plugin Properties Options

- Please refer to Plugin Properties Options for more information.

Tutorials

- Guideline for Developing a Plugin
- How to develop a Userview Theme Plugin

Related Community Plugins

- **Corporati Theme**

  org.joget.plugin.enterprise.CorporatiTheme (wflow-core)

- **V3 Default Theme**

  org.joget.apps.userview.lib.DefaultTheme (wflow-core)

- **V6 Universal Theme**
  org.joget.plugin.enterprise.UniversalTheme (wflow-core)
- **DX Progressive Theme**
  org.joget.plugin.enterprise.ProgressiveTheme (wflow-core)