Custom Password Reset Tool in SAP Enterprise Portal Using Web Dynpro for Java

Applies to:
SAP Enterprise Portal, Web Dynpro for Java. For more information, visit the Portal and Collaboration homepage.

Summary
This article is all about developing a new password reset tool for SAP Enterprise portal using Web Dynpro for Java. This simple application can be made available to the portal end users to reset their passwords themselves without requesting administrator. This tool can be reused in any project when SAP Enterprise Portal as a frontend and UME Data source is writable.

Author: Ramganesan Karuppaiah
Company: Infosys Technologies Limited, Bangalore, India
Created on: 26 November 2010

Author Bio
Ramganesan Karuppaiah is a NetWeaver Portal Consultant currently working with Infosys Technologies Limited, Bangalore. He has 4.5+ years of extensive experience in SAP Enterprise Portal and ABAP. He is an SAP Certified ABAP with NetWeaver 7.0 development Associate. Also he is a Sun Certified Java Programmer 1.5. His nature of work involves Portal Administration, Portal Customization, Personalization and Content Development & Customization using Web Dynpro for Java, JSP Dynpage and ABAP.
# Table of Contents

Introduction ........................................................................................................................................ 3
Preview of Output ................................................................................................................................ 3
Procedure ........................................................................................................................................... 3
  Creating Web Dynpro Project ........................................................................................................ 3
  Creation of Context Elements ...................................................................................................... 6
  Creation of methods ....................................................................................................................... 8
  UI changes ...................................................................................................................................... 10
  Implementation of Methods ........................................................................................................ 10
  Deploy and Run ............................................................................................................................ 14
Advantages and Drawbacks ............................................................................................................. 14
  Advantages ................................................................................................................................... 14
  Drawback ..................................................................................................................................... 14
Related Content .............................................................................................................................. 15
Disclaimer and Liability Notice ...................................................................................................... 16
Introduction

The document is about developing a new password reset tool for SAP Enterprise portal using Web Dynpro for Java. This application can be made available to the portal end users to reset their passwords themselves without requesting administrator.

Preview of Output

The completed Web Dynpro application shall display the end result as per the screen shot below.

![Change Password Form](image)

Procedure

Below are the steps required to create and run the password reset applications in Web Dynpro for Java.

- Creation of Web Dynpro Project
- Creation of Context Elements
- Creation of methods
- UI Changes
- Implementation of Methods
- Deploy and Run

The below details are used in this article to describe the steps.

<table>
<thead>
<tr>
<th>Web Dynpro Project Name:</th>
<th>PasswordReset_WD</th>
</tr>
</thead>
<tbody>
<tr>
<td>Component Name:</td>
<td>PasswordReset_WDCmp</td>
</tr>
<tr>
<td>View Name:</td>
<td>PasswordResetView</td>
</tr>
<tr>
<td>Application Name:</td>
<td>PasswordResetApp</td>
</tr>
</tbody>
</table>

Creation of Web Dynpro Project

Open Web Dynpro Perspective in NWDS → File → New → Web Dynpro Project
Enter Web Dynpro Project Name → Press Finish

Right click on Application to create an Application → Create Application

Enter Application and package details
Select Create a New component → next

Provide all required details.
Creation of Context Elements

Open View Context → Create context elements as per the below screen shot.

Right click on the View → Apply Template
Select Form → Next

Select the context element → next → Finish

A form shall be created in the view PasswordResetView as per the above screenshot.
Creation of methods

Go to Methods Tab to create methods -> Press New to create a new method.

Create 2 methods as per the below screenshot details. The method clearForm shall be used to reset the form data entered by end users. The method resetPassword shall perform validations and change the password upon valid inputs.
Create Action button ➔ Press Next

Select the event Handler properties for Password reset button ➔ Finish
UI changes

Change the look and feel of the layout as per the below details
1. Set the input field “User Name” as read only.
2. Set all the fields are mandatory
3. Set Old Password, New Password and Confirm New Password fields are as password fields.

Implementation of Methods

Below validations can be used in this password change applications.

- Check for All Mandatory fields.
- Check for confirm and new passwords’ equality.
- Check for old and new passwords’ inequality.
- The New password should not be the used id.
- The first letter of the password should be an alphabet.
- New password length should be between 6 and 12 chars.
- Check for invalid old password.
- Change the password if the given input is valid.

UME API methods have been used is used to compare the old password entered by enduser with LDAP. Also, it changes the password on valid inputs.
public void resetPassword()
{
//@@begin resetPassword()

String userId = wdContext.currentContextElement().getUserId();
String oldPass = wdContext.currentContextElement().getOldPassword();
String newPass = wdContext.currentContextElement().getNewPassword();
String confPass = wdContext.currentContextElement().getConfirmNewPassword();
boolean isValidPwd = false;

if(userId != null && !userId.equals("") && oldPass != null && !oldPass.equals("")) { //Check for Mandatory Fields

    if(confPass.equals(newPass)) {  // Confirm password and New Password should be same
        if(!oldPass.equalsIgnoreCase(newPass)) { // The New password should not be the old password
            if(!newPass.equalsIgnoreCase(userId)) { // The New password should not be the user id
                char start = newPass.charAt(0);
                // The first letter of the password should be an alphabet
                if((start >= 'a' && start <= 'z') || (start >= 'A' && start <= 'Z')) {  
                    if(newPass.length() >= 6 && newPass.length() <= 12) { // New password length should be between 6 and 12 chars
                        try{
                            IUserAccount userAcct = UMFactory.getUserAccountFactory().getUserAccountByLogonId(userId);
                            IUserAccount mutUserAcct = UMFactory.getUserAccountFactory().getMutableUserAccount(userAcct.getUniqueID());

                                // Check for valid password
                                isValidPwd = userAcct.checkPassword(oldPass);  
                                if(isValidPwd){
                                    userAcct.setPassword(newPass); // Change the Password
                                    userAcct.commit(); // Save the change
                                wdComponentAPI.getMessageManager().reportSuccess("Your Password has been changed now!");
                                wdContext.currentContextElement().setNewPassword("");
                                wdContext.currentContextElement().setConfirmNewPassword("");
                        
                        }  
                    }  
                }  
            }  
        }  
    }  
}  
}  
}
wdContext.currentContextElement().setOldPassword(""); 
} 
else{
  wdComponentAPI.getMessageManager().reportException("Invalid Old Password!",true);
  wdContext.currentContextElement().setOldPassword("");
} 
}

catch(UMException e){
  wdComponentAPI.getMessageManager().reportException("The Password can not be changed!",true);
} 
} else{
  wdComponentAPI.getMessageManager().reportException("The password length should be between 6 and 12 chars!",true);
  wdContext.currentContextElement().setNewPassword("");
  wdContext.currentContextElement().setConfirmNewPassword("");
} 
} else{
  wdComponentAPI.getMessageManager().reportException("New Password should start with an alphabet", true);
  wdContext.currentContextElement().setNewPassword("");
  wdContext.currentContextElement().setConfirmNewPassword("");
} 
} else{
  wdComponentAPI.getMessageManager().reportException("User Id can not be a password!",true);
  wdContext.currentContextElement().setNewPassword("");
  wdContext.currentContextElement().setConfirmNewPassword("");
} 
}

else{
  wdComponentAPI.getMessageManager().reportException("Old Password and New Password should not be same!",true);
  wdContext.currentContextElement().setNewPassword("");
  wdContext.currentContextElement().setConfirmNewPassword("");
  wdContext.currentContextElement().setOldPassword("");
The method `clearForm()` clears the inputs fields entered by the end user i.e. It resets the form.

```java
public void clearForm()
{
    //@@begin clearForm()  
    wdContext.currentContextElement().setNewPassword("" ); 
    wdContext.currentContextElement().setConfirmNewPassword("" ); 
    wdContext.currentContextElement().setOldPassword("" ); 
    //@@end
}
```
Deploy and Run

Create Web Dynpro Application → Deploy New Archive and Run. The Application is launched in a new browser Window.

To enable the deployed application to portal end-users, a new iView can be created against the deployed Web Dynpro application and displayed in portal as a part of Detailed Level Navigation/Top Level Navigation/a link on masthead to access the application.

Advantages and Drawbacks

Advantages

1. Simple, Handy and User friendly.
2. Enables end users to reset the portal password.
3. Only authorized persons (i.e. Logged in user) can change the password.
4. This tool uses standard UME API to validate the current password and change it as per the user inputs.
5. This tool is applicable to all portal versions.
6. This can be reused in any project when SAP Enterprise portal as a front end and UME Data source is writable.

Drawback

1. The password can not be changed if the UME Data source is read only. This scenario might occur when the corporate LDAP is the data source for portal.
Related Content

Creating a Simple Web Dynpro Java Application
Creating your first Web Dynpro Java Application
Other Web Dynpro references
For more information, visit the Portal and Collaboration homepage.
Disclaimer and Liability Notice

This document may discuss sample coding or other information that does not include SAP official interfaces and therefore is not supported by SAP. Changes made based on this information are not supported and can be overwritten during an upgrade.

SAP will not be held liable for any damages caused by using or misusing the information, code or methods suggested in this document, and anyone using these methods does so at his/her own risk.

SAP offers no guarantees and assumes no responsibility or liability of any type with respect to the content of this technical article or code sample, including any liability resulting from incompatibility between the content within this document and the materials and services offered by SAP. You agree that you will not hold, or seek to hold, SAP responsible or liable with respect to the content of this document.