



PDF SHARE FORMS

Online, Offline, OnDemand

PDF forms and SharePoint are better together

User Profile Service usage with PDF Share Forms

Product: PDF Share Forms Enterprise for SharePoint 2013

This guide describes how to use User Profile Service with PDF Share Forms –
retrieve information from User Profile by C# scripting.

Preparations

Before designing a template, make sure that following assemblies are added in **Central Administration → PDF Share Forms → Scripting References**. See Figure 1.

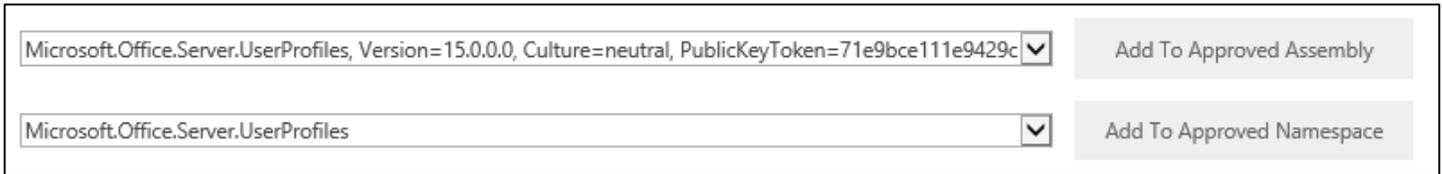


Figure 1. Adding new assembly and namespace

In addition, check names in User Profile properties. Later in the guide, property names are used in scripting. Therefore, it is important to check internal names in User Profile properties.

Navigate to **Central Administration → Application Management → Manage service applications → User Profile Service Application → Manage User Properties**. There is a list of all User Profile properties. See Figure 2.

> Contact Information	↑ ↓	Section
Work email	↑ ↓	E-mail
Mobile phone	↑ ↓	string (Single Value)
Fax	↑ ↓	string (Single Value)
Home phone	↑ ↓	string (Single Value)
Office	↑ ↓	string (Single Value)
Office Location	↑ ↓	string (Single Value)
Assistant	↑ ↓	Person

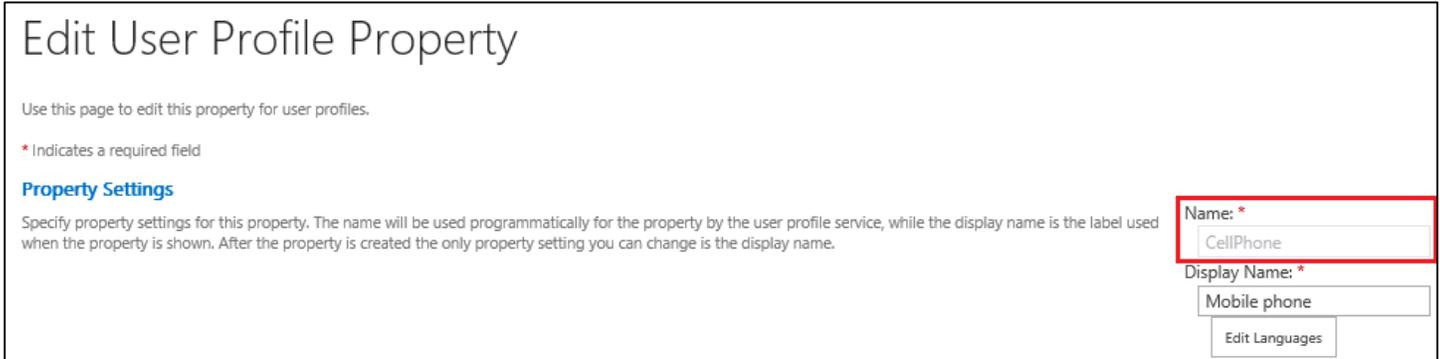
Figure 2. User Profile property list

In this step “Mobile phone” property is used as an example. Click on “Mobile phone” and choose “Edit”. As seen in Figure 3.

> Contact Information	↑ ↓	Section
Work email	↑ ↓	E-mail
Mobile phone	↑ ↓	string (Single Value)
Fax	↑ ↓	string (Single Value)
Home phone	↑ ↓	string (Single Value)

Figure 3. User Profile list – edit property

As it can be seen in Figure 4 internal name can be different from display name.



The screenshot shows the 'Edit User Profile Property' interface. The title is 'Edit User Profile Property'. Below the title, it says 'Use this page to edit this property for user profiles.' and '* Indicates a required field'. There is a section for 'Property Settings' with instructions: 'Specify property settings for this property. The name will be used programmatically for the property by the user profile service, while the display name is the label used when the property is shown. After the property is created the only property setting you can change is the display name.' On the right side, there are three input fields: 'Name: *' with the value 'CellPhone', 'Display Name: *' with the value 'Mobile phone', and an 'Edit Languages' button.

Figure 4. Edit User Profile "Mobile Phone" property

For the purposes of scripting internal property name should be used instead of the display name.

User Profile Example

Figure 5 shows what information is retrieved from user profile and used in form.



The screenshot shows a user profile for 'John Smith'. It includes a 'CHANGE YOUR PHOTO' button, a profile picture placeholder, and the name 'John Smith'. Below the name, there is a 'Newsfeed' button and 'John Smith's Activities' with a help icon. The main content area says 'About John Smith' and 'edit your profile'. Below that, it says 'Feel free to mention me in a post or ask me about Development.' On the right side, there is a table of user information:

Mobile phone	+1 888 792 7871
Skills	C++ JavaScript C#
Birthday	July 30

Below the table is a 'SHOW LESS' button.

Figure 5. User Profile information for specific SharePoint user

Template

Step 1. Preparation of a template

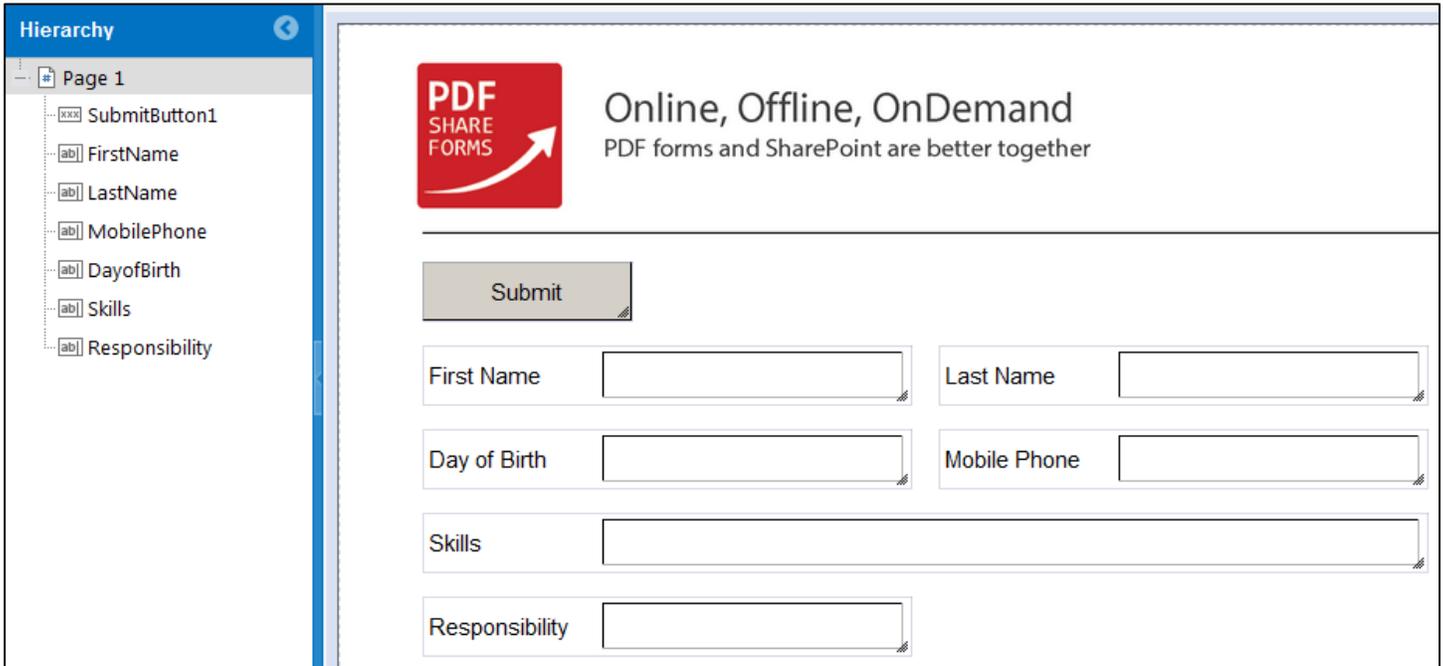


Figure 6. Designed template

Following controls are used in the template as seen in Figure 6:

- SubmitButton1 – submitting the form to a SharePoint library;
- Text Fields – “First Name”, “Last Name”, “Day of Birth”, “Mobile Phone”, “Skills” and “Responsibility”.

These text fields are populated with the information from User Profile.

Step 2. Add form load script. Navigate to **Developer tab** → **Form Load**

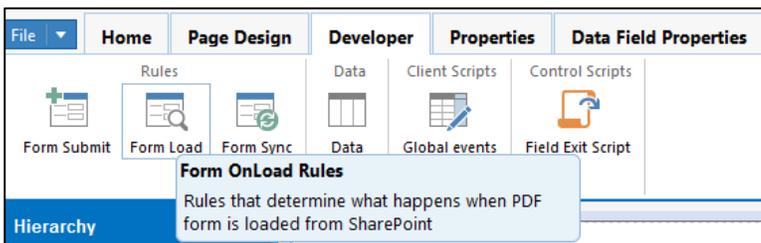


Figure 7. Form Load rules

In Form Load Rules menu add **“Execute C# script”** action below the **“Run system OnLoad”** action

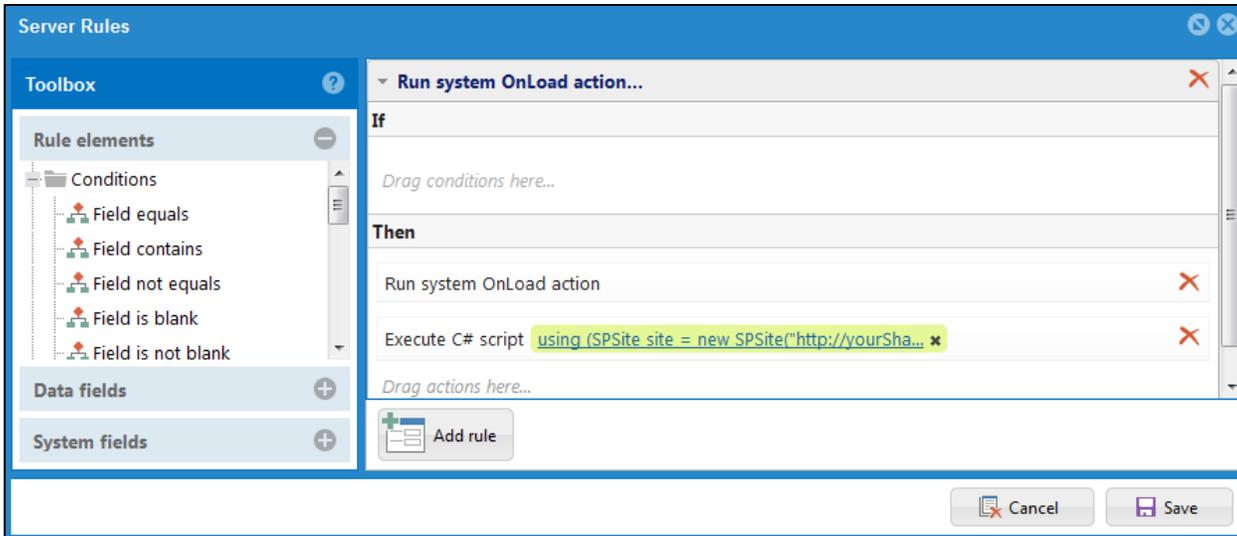


Figure 8. Form Load rules window

Script

```
using (SPSite site = new SPSite("http://yourSharePointSite")) //SharePoint site URL must be
specified
{
    SPServiceContext serviceContext = SPServiceContext.GetContext(site);
    // Create a user profile that uses the default user profile subtype.
    UserProfileManager userProfileMgr = new UserProfileManager(serviceContext);
    UserProfile userProfile = userProfileMgr.GetUserProfile(currentWeb.CurrentUser.LoginName);
    //get user properties from user profile; make sure that internal property names are called
    data.resolveNode("FirstName").Value = userProfile["FirstName"].Value.ToString();
    data.resolveNode("LastName").Value = userProfile["LastName"].Value.ToString();
    data.resolveNode("MobilePhone").Value = userProfile["CellPhone"].Value.ToString();
    data.resolveNode("DayofBirth").Value = Convert.ToDateTime(userProfile["SPS-
    Birthday"].Value).ToString("MMM dd");
    data.resolveNode("Skills").Value = userProfile["SPS-Skills"].Value.ToString();
    data.resolveNode("Responsibility").Value = userProfile["SPS-
    Responsibility"].Value.ToString();
}
```

This script contains connection to a SharePoint site where form is stored and connection to the User Profile for the current user. Then specific fields are populated with the information from the current user’s User Profile.

Step 3. Deploy template to a library



Runtime

Navigate to deployment library and create new form

The screenshot shows a web form with the following elements:

- Header:** PDF SHARE FORMS logo and the text "Online, Offline, OnDemand" and "PDF forms and SharePoint are better together".
- Submit Button:** A grey button labeled "Submit".
- Form Fields:**
 - First Name:** John
 - Last Name:** Smith
 - Day of Birth:** Jul 30
 - Mobile Phone:** +1 888 792 7871
 - Skills:** C++, JavaScript, C#
 - Responsibility:** Development

Figure 9. Form in runtime

All fields are populated with the information via form load from current user profile.