

Company Visitor Pre-Registration App Setup Guide

This guide explains how to install and configure the Visitor Registration App in your company’s Microsoft 365 tenant.

CONTENTS

- Step 1: Prepare the SharePoint List2**
 - OPTION 1 (Using Powershell Script)2
 - OPTION 2 (Manually created in SharePoint Site)3
- Step 2: Import the Power Apps Package3**
- Step 3: Reconnect Data Sources4**
- Step 4: Publish the App5**
- Step 5: Test the App5**
- Configure the App Logic6**

Step 1: Prepare the SharePoint List

OPTION 1 (Using Powershell Script)

Download the Powershell script “createsplist_PreReg.ps1” ([Download Script](#))

- close all opening Powershell sessions
- run powershell in administrator mode
- run Get-InstalledModule to check installed module version
- If SharePointPnpPowerShellOnline exists in any other version than 3.23.2007.1, run Uninstall-Module SharePointPnpPowerShellOnline
- run Install-Module -Name SharePointPnPPowerShellOnline -RequiredVersion 3.23.2007.1
- run Import-Module SharePointPnPPowerShellOnline -Verbose
- modified script createsplist_MFA.ps1 (\$SiteName, \$ListName, \$Domain) and run with sharepoint admin login account.
- if an error "The remote server returned an error: (403) Forbidden." make sure you use admin account, and run Disconnect-PnPOnline before running the script again

\$SiteName: Give a new name that will be used to create a new Sharepoint site to store the Visitor Registration data list.

\$ListName: Give a new name that will be used as the name of the list storing the Visitor Registration data

\$Domain: State the domain name used for your organization and in Sharepoint

The script will create columns in the List which are

FirstName, LastName, Email, Company, LicensePlate,
AppointmentDateTime, PersonToVisit, PersonToVisitID, PersonToInformAboutVisit,
PersonRegisteringVisit

Check if List created successfully by going to

<https://{domain}.sharepoint.com/sites/{siteName}/Lists/{listName}>

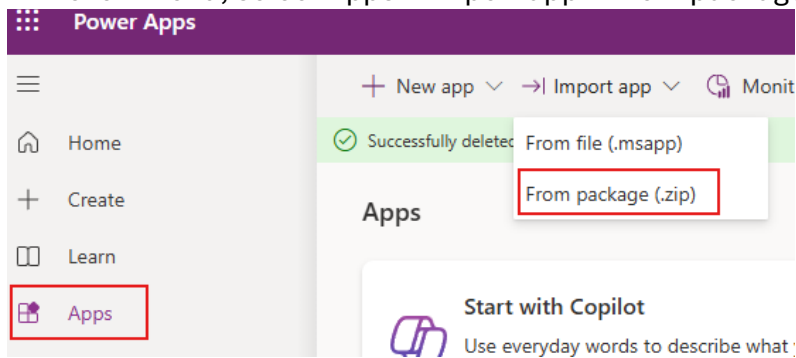
OPTION 2 (Manually created in SharePoint Site)

Create a new list called < "HLPreReg" or create your own SharePoint List Name> with the following columns:

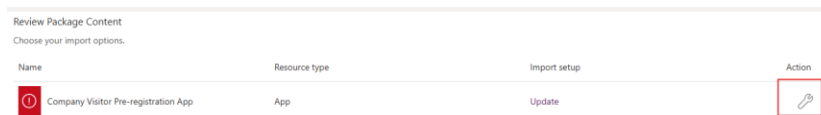
Column Name	Type	Notes
Title	Single line text	Required by SharePoint (can be hidden)
FirstName	Single line text	Visitor's First name
LastName	Single line text	Visitor's Last name
Email	Single line text	Visitor's Email
Company	Single line text	Visitor's Company
LicensePlate	Single line text	Visitor's License Plate
AppointmentDateTime	Date & Time	Include Time = Yes
PersonToVisit	Single line text	Stores DisplayName from AAD
PersonToVisitID	Single line text	Stores Email from AAD
PersonToInformAboutVisit	Single line text	Stores Email from AAD
PersonRegisteringVisit	Single line text	Auto-filled with logged-in user

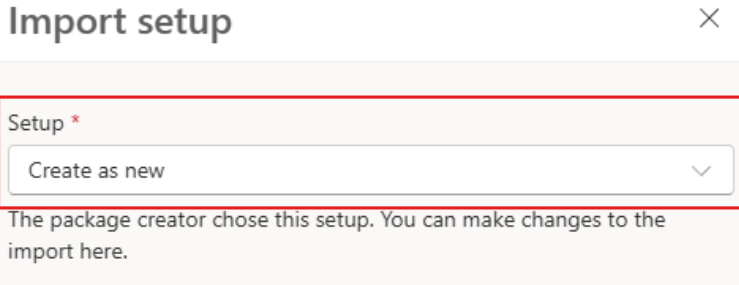
Step 2: Import the Power Apps Package

- Go to Power Apps Maker Portal. (<https://make.powerapps.com/>)
- In the left menu, select Apps → Import app → From package (.zip)



- Upload the downloaded .zip package file.
- Follow the wizard → choose Create as new.

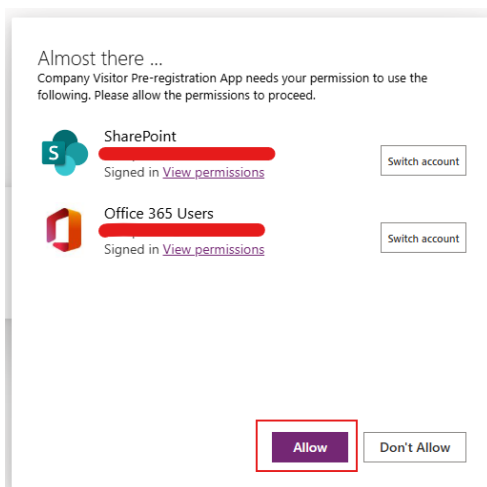




- Save and finish the import.

Step 3: Reconnect Data Sources

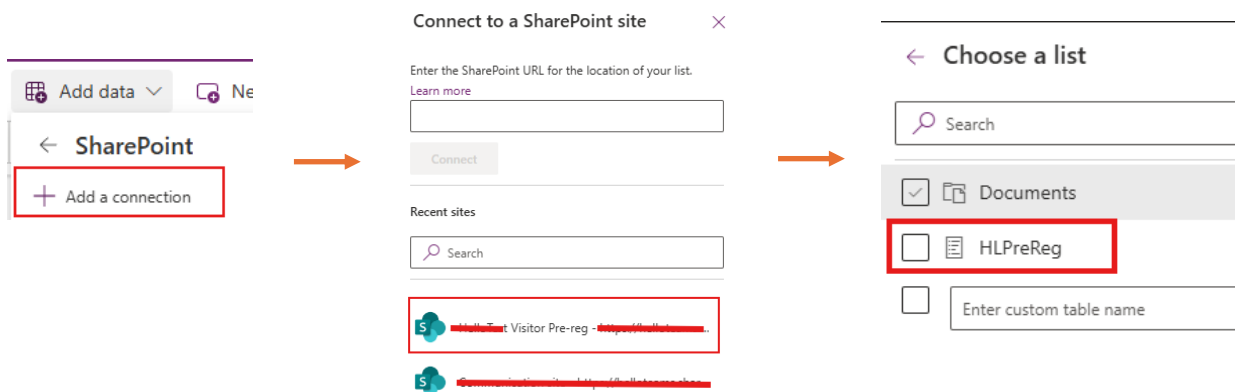
Open and allow the App.



Reconnect to SharePoint and Office 365 Users:

In the app editor:

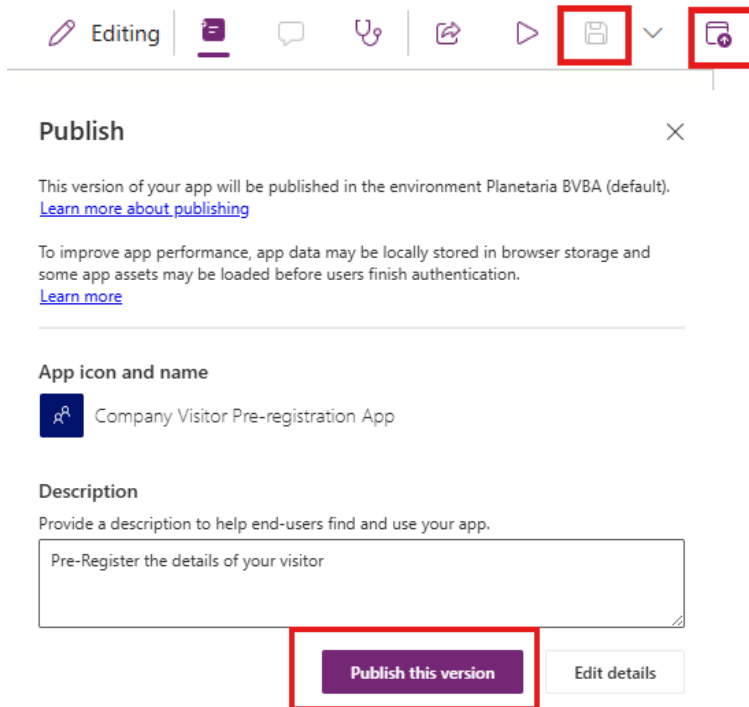
- SharePoint → Delete the existing SharePoint connection and point it to the new "SharePoint List" on your environment.



- Office 365 Users → allows lookup of employees in Azure AD.
- If the SharePoint List isn't named "**HLPreReg**", you must map the formula for each field within the app editor. Navigate to the "[Configure App Logic](#)" section for guidance.

Step 4: Publish the App

- In the app editor → Save.
- Then Publish → Publish this version.



Step 5: Test the App

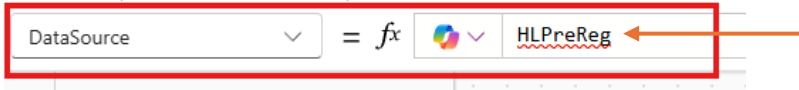
- Open the app from Power Apps portal or mobile app.
- Register a test visitor.
- Verify the entry in SharePoint (Name, Email, Appointment Date/Time).

End Result: The app will allow visitor pre-registration, automatically fill people from Azure AD, save to SharePoint, and work across desktop and mobile.

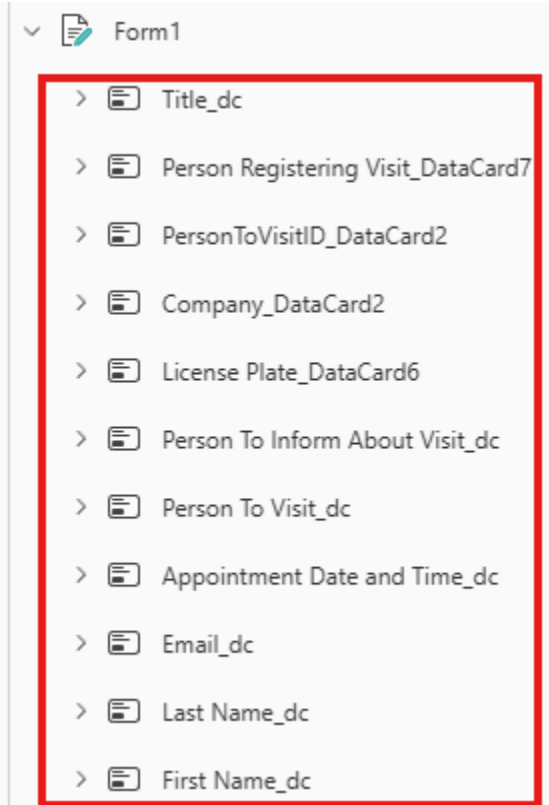
Configure the App Logic

Change the formulas for each Fields:

- Form 1 → DataSource: : *Change it to <Your SharePoint List Name>*

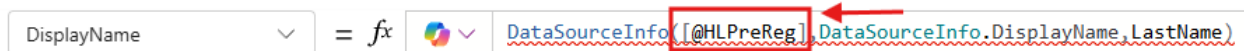


- For all DataCards change the formula of “**DisplayName**” and “**Maxlength**”



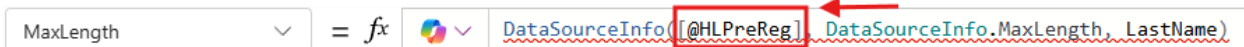
Example: LastName → DataCard

DisplayName: Change the DataSourceInfo to your SharePoint List



Example: LastName → DataCard

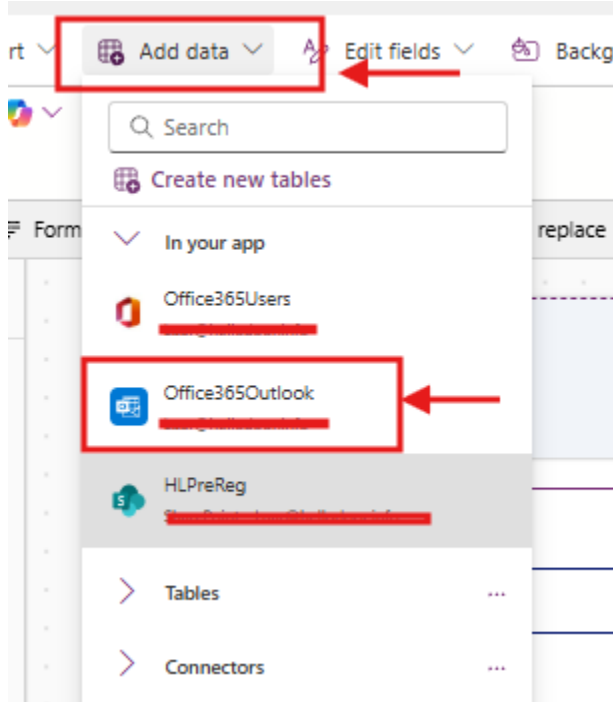
Maxlength: Change the DataSourceInfo to your SharePoint List



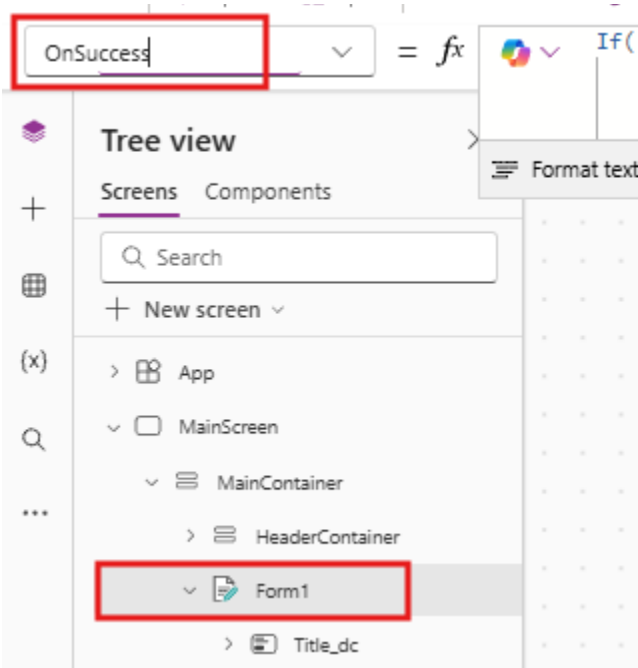
- Apply the changes to each DataCard.

Enabling Visitor's Email Send Notification

- Add data connection and choose Office365Outlook



- Change the formula of Form1 – OnSuccess Property

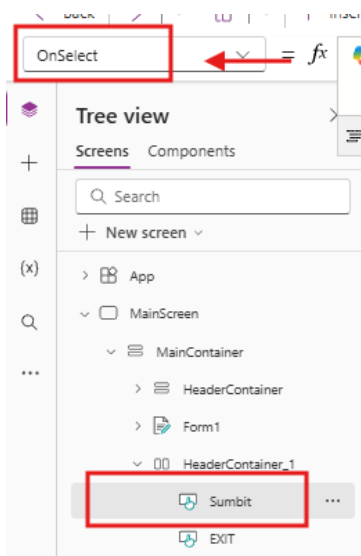


```

If(
  !IsBlank(varVisitorEmail),
  Office365Outlook.SendEmail(
    varVisitorEmail,
    "Your Visit Registration Confirmation",
    "<p>Hello " & Form1.LastSubmit.FirstName & " " & Form1.LastSubmit.LastName &
    "</p>" &
    "<p>Thank you for registering your visit to <b>HelloDoor</b>.</p>" &
    "<p><b>Appointment Details:</b><br>" &
    "Date and Time: " & Text(Form1.LastSubmit.AppointmentDateTime, "dddd, mmmm dd,
    yyyy") & " at " & Text(Form1.LastSubmit.AppointmentDateTime, "hh:mm AM/PM") & "<br>"
    &
    "License Plate: " & Form1.LastSubmit.LicensePlate & "<br>" &
    "Person to Visit: " & Form1.LastSubmit.PersonToVisit & "</p>" &
    "<p>We look forward to your visit!</p>" &
    "<p>Best regards,<br>" & User().FullName & "</p>",
    {IsHtml: true}
  );
  Notify("✔ Visitor pre-registration submitted and email sent!", NotificationType.Success);
  ResetForm(Form1);
  NewForm(Form1);
  Navigate(Success, ScreenTransition.Fade),
  Notify("⚠ Email address missing — form saved but no email sent.",
  NotificationType.Warning)
)

```

- Change the formulas of Submit– OnSelect Property



```

If(
  !IsMatch(DataCardValue3.Text, "^[A-Za-z0-9._%+-]+@[A-
  Za-z0-9.-]+\.[A-Za-z]{2,}$"),
  Notify("Please enter a valid email address before
  submitting.", NotificationType.Error),
  Set(varVisitorEmail, DataCardValue3.Text);
  SubmitForm(Form1)
)

```