Microsoft Power platform
ISV Partner Getting Started Guide
**Introduction**

Power Platform for Dynamics 365 offers a powerful, point-and-click approach to app building that makes it easy for anyone familiar with Microsoft Office to customize and extend Dynamics 365 and Office 365 and to build a new category of apps.

This guide is broken down into four sections: First, it outlines the process to become a Microsoft partner, including registering for the Microsoft Partner Network (MPN) and Cloud Partner Portal (CPP). If you are already a Microsoft partner with Cloud Partner Portal (CPP) access, you can skip this section. The second section provides an overview of Power platform, and the third section provides guidance on building apps with Power platform. The fourth section walks through the steps to share your app, as well as how to package and publish your application for sale on AppSource, including information on Microsoft’s Go-to-market services. Lastly, we’ve included links to additional resources to help you find answers to questions and provide additional guidance for when you’re ready to move beyond the basics.
Contents

4 Prerequisites
Becoming a Microsoft partner
Register as a publisher
Create a seller account
Sign up for Power platform

8 Understanding the Power platform
Parts of the Power platform
Common Data Service
Model-driven apps vs. Canvas apps
Ways to build PowerApps
Programming models

14 Building your application
Create a model-driven app
Create a canvas app
Build with Power BI and Microsoft Flow

21 Publish your PowerApp to AppSource
Create a managed solution package
Validate your code
Create a new offer
Submit for publication
Publish your solution
Market and sell

29 Additional resources
Prerequisites

Step 1  
Become a Microsoft partner  
• Create a Microsoft work account  
• Join the Microsoft Partner Network  
• Set up your Partner Center account

Step 2  
Register as a publisher  
• Register in the Cloud Partner Portal

Step 3  
Create a seller account  
• Create a Microsoft Developer account

Step 4  
Sign up for Power platform  
• Register for a PowerApps account  
• Sign up for the Dynamics Insider Program

If you are already a Microsoft Partner with Cloud Partner Portal access, you can skip this section.
Prerequisites

Before you begin building and publishing your first apps, there are a few steps you must take to get set up as a Microsoft Partner. These include joining the Microsoft Partner Network (MPN) and getting registered in the programs that enable you to publish, market, and sell your apps.

1 Become a partner

Becoming a Microsoft partner gives you access to the Microsoft resources you need to build, market, and sell your solution. While you don’t need to be a Microsoft partner to begin developing your apps, all of the steps below are required to gain access to the programs that enable you to publish, market, and sell your apps.

Create a Microsoft work account
Before you begin, you must create a Microsoft work account. The same account should be used as you sign up for subsequent programs to ensure all of your privileges are centralized under a single account ID. You can register your email at https://signup.live.com.

Join the Microsoft Partner Network
Becoming a Microsoft partner gives you access to all the resources you need to build and publish apps. To become a partner, you need to join the Microsoft Partner Network (MPN), at which time you will be assigned an MPN ID. MPN membership is free to all partners, and you can enroll in MPN at https://partner.microsoft.com/en-us/membership.

If you have an active subscription to Microsoft Azure or Office 365, you already have a Microsoft work account.
Prerequisites

Set up your Partner Center account
Once you have joined the Microsoft Partner Network (MPN), you can set up your Partner Center account. Your Partner Center account provides you with access to pricing information, tools and services, and enables you to manage admin credentials for your company’s work account. Partner Center is also where you can purchase or renew subscriptions to Microsoft Action Packs, create a business profile to receive and manage sales leads from Microsoft, and see if you qualify for co-selling opportunities.

Register as a publisher
Registering as a publisher allows you to sell your solutions on AppSource, the marketplace that gives ISVs access to more than 100 million commercial users.

Register in the Cloud Partner Portal
The first step to becoming a publisher is to register in the Cloud Partner Portal (CPP). The Cloud Partner Portal (CPP) is where you submit your apps for publication, publish and promote your apps, and manage your Dynamics 365 for Customer Engagement offers.

To begin the registration process, you must complete a brief form located at https://appsource.microsoft.com/en-us/partners/signup.

Shortly thereafter, one of our team members will follow up to help you complete your registration. Once registered, you can access the Cloud Partner Portal (CPP) at https://cloudpartner.azure.com.

Learn how to manage your Partner Center account.

If you are an Azure partner already registered in the Cloud Partner Portal, you don’t need to register again; however, we recommend you submit the signup form to connect with a member of our Customer Engagement program management team.

In the future, you will be able to select PowerApps. Currently, you must select Customer Engagement.
3 Create a seller account

Creating a seller account gives you the necessary resources to market and sell your solutions on AppSource.

Create a Microsoft Developer account
To market and sell your solutions on the AppSource marketplace, you must create a Microsoft Developer account in the Microsoft Developer Center at http://dev.windows.com/registration. When registering, be sure to use the same Microsoft account ID that was used for the Cloud Partner Portal to ensure that your Microsoft Developer and Cloud Partner Portal accounts are appropriately linked.

4 Sign up for Power platform

Register for a Power platform account
To begin building PowerApps, you must set up an account. You can try PowerApps for free by signing up either for a 30 day trial or community plan.

Sign up for the Dynamics Insider Program
In addition to the above steps, all partners must join the Dynamics Insider Program, which you can find at https://experience.dynamics.com/insider. Through the Dynamics Insider Program, you can gain access to partner resources, test and validate new features, and provide your valuable feedback.
Understanding the Power platform

Step 1
Parts of the Power platform
- PowerApps
- Microsoft Flow
- Power BI

Step 2
Common Data Service

Step 3
Model-drive apps vs Canvas apps
- Model-driven apps
- Canvas apps

Step 4
Ways to build PowerApps
- Check out some sample apps
- Create an app from a template
- Use shared apps
- Create an app from a data source
- Build from the ground up

Step 5
Programming models
- Build
- Extend
- Connect
Microsoft’s Power platform is comprised of three tools—PowerApps, Flow, and Power BI—designed to help you build, extend, and connect apps for Dynamics 365 and Office 365.

### PowerApps
PowerApps sits at the center of the Power platform. PowerApps is a collection of services, apps, and connectors that work together to let you build applications ranging from simple no-code mobile apps used to view and update your data to fully featured software that extends the functionality of our Dynamics 365 solutions. With PowerApps, you do much more than just view your data. You can act on your data and update it anywhere and from any device.

Sign up for a free trial version and sign-in at [http://powerapps.com](http://powerapps.com) and explore samples and templates.

### Building blocks
To create, share, and administer apps, you’ll use these sites:

1. **web.powerapps.com**: In this site, you can open apps, specify the type of app that you want to create, and create data connections and flows. To use this site you’ll need to log in using your organizational account.

2. **PowerApps Studio**: In this site, you build apps by configuring user interface (UI) elements and Excel-like formulas.

3. **PowerApps Mobile**: Run your apps on Microsoft Windows, Apple iOS, and Google Android devices.

4. **PowerApps admin center**: In this site, you’ll define environments and data policies.

### Related technologies
Microsoft PowerApps works with other technologies to help you build and share apps across your organization. Let’s take a look at some of these technologies:
Understanding the Power platform

- **Microsoft Dynamics 365**: Dynamics 365 is the home for all your business apps: Microsoft Power BI, Microsoft Flow, Dynamics 365 for Talent, and many others.
- **Microsoft AppSource**: AppSource is where you share apps and download other apps that might be useful to your business.
- **Data sources**: Without data, you don’t have a business. Data sources bring cloud and on-premises data into your apps. You access data through built-in connections, custom connectors, and gateways.

**Microsoft Flow**
With Microsoft Flow, users can connect to more than 200 services to create automated multi-step workflows and easily create custom connections when needed. Once built in Flow, these workflows and connections can be leveraged to extend the functionality of your PowerApps apps. Sign-in at http://flow.microsoft.com and explore templates.

**Power BI**
Power BI enables users to build sophisticated, visual dashboards from their data. As part of the Power platform, Power BI dashboards components can easily be embedded into PowerApps apps, and PowerApps components can be embedded into Power BI dashboards. Sign-in at http://powerbi.microsoft.com and explore templates.

**Build and certify custom connectors**
Bring your product into the Microsoft cloud by building a connector that enables PowerApps to talk to your service.

**Common Data Service**
Microsoft’s Business Applications—including Dynamics 365, Office 365, and Power platform—are all built on top of our Common Data Service. Common Data Service for Apps lets you securely store and manage data used by business applications. Data within Common Data Service is stored within a set of records called entities. An entity is a set of records used to store data, similar to how a table stores data within a database.

The Common Data Service includes a base set of standard entities that support common business scenarios that connect to Dynamics 365 application data. You can also create custom entities specific to your organization and populate them with data that you import from lists in SharePoint, from Excel, or from PowerQuery. App makers can then use PowerApps to build rich applications using this data.

Dynamics 365 applications, including Dynamics 365 for Sales, Service, and Talent, use Common Data Service for Apps to store and secure data used by the applications. This means you can build apps using PowerApps and Com-
Understanding the Power platform

Data Service for Apps directly against your core business data already used within Dynamics 365 without the need for manual integration. Dynamics 365 for Finance and Operations and Dynamics 365 for Retail currently require the configuration of the Data Integrator to make your business data available within Common Data Service for Apps.

Model-driven apps vs Canvas apps
There are two types of apps you can build within PowerApps: Model-driven apps and Canvas apps.

Model-driven apps
Model-driven app design is an approach that focuses on adding dashboards, forms, views, and charts to your apps. With no code, or very little, you can build apps that are simple or very complex.

Canvas apps
Canvas apps allow users to design and build business apps without writing code in a traditional programming language. The app can be designed by dragging and dropping elements onto a canvas, just as you would design a slide in PowerPoint. Excel-like expressions can be used for specifying logic and working with data, and apps can integrate business data from a wide variety of Microsoft and third-party sources.

In canvas apps, the designer has total control over the app layout. In model-driven apps, on the other hand, much of the layout is determined by the components you add. The emphasis is more on quickly viewing your business data and making decisions than on intricate app design.

Ways to build PowerApps
There are several different ways you can create an app, including from a sample app, a template, a shared app, or a data source.
Understanding the Power platform

Check out some sample apps
On https://web.powerapps.com, you’ll be greeted with several sample apps that you can use in your web browser. Explore these apps to quickly get a sense of what’s possible and how PowerApps can help your business. These samples also contain sample data to inspire your thinking. For example, this app can help service representatives for a flooring company give accurate and immediate cost estimates when they visit customer locations.

Create an app from a template
Another good way to create an app is to start from a template. Templates use sample data to help you get a sense of what’s possible. By opening them in PowerApps Studio, you can learn hands-on how an app is built.

Use shared apps
If your team has started to use PowerApps, others might have shared apps with you already. If the app maker gave you permission to edit the app, it appears on web.powerapps.com, where you can open, customize, and share the app.

Create an app from a data source
A great way to get started is to generate an app from your own data. Just point PowerApps at the data source of your choice, and watch as PowerApps automatically builds a three-screen app. Data sources may include the Common Data Service, SharePoint, and Excel.

Build from the ground up
You can also build an app from scratch and add all the pieces as you go. You can then branch out and let your imagination run wild. There will be plenty of that later in this module! The easiest way to learn about PowerApps is to start with a sample app, open a template, or connect to your data source and have PowerApps generate the app for you.
Understanding the Power platform

Programming models
There are three basic programming models that you can use to build your application: build, extend, connect.

Build
ISVs can build solutions that embed transactional data and analytical insights into business apps. These solutions come in many flavors and may include embedding Power BI dashboards into a multi-tenant app or building a Power-Apps canvas app that is embedded into Dynamics 365 or Power BI.

Extend
ISVs can enhance Dynamics 365 apps by building new IP into the platform and/or extending the Common Data Model. This may include extending horizontally—such as designing a new search UI or building new custom controls that add capabilities, like a signature pad, that can be leveraged across applications—or extending vertically—such as customizing a version of Dynamics 365 for Marketing for veterinary clinics. ISVs can also extend the platform by defining new entities in the Common Data Model that can then be leveraged by other ISVs.

Connect
Your users may need to connect to data that is living in another system. Our platform currently enables you to connect data and processes from over 200 external systems to help you build hybrid end-to-end solutions. For circumstances when a connector doesn’t already exist, ISVs can build custom connectors using our no-code/low-code experience, including connectors to their own systems, and these connectors can be sold directly in AppSource or as part of a larger solution.

Watch the session from Microsoft Build 2018 and learn how to accelerate your SaaS App development using the power of the Business Application Platform
Building your application

Step 1
Create a canvas app
- Create a canvas app
- Create accessible canvas apps
- Customize your app
- Use App checker to fix errors and make accessible apps
- Sharing canvas apps
- Manage a canvas app

Step 2
Create a model-driven app
- Building blocks of model-driven apps
- Design phases
- Create your model-driven app
- Customize your apps
- Use App checker to fix errors and make accessible apps
- Sharing model-driven apps
- Managing the properties of a model-driven app

Step 3
Build with Power BI and Microsoft Flow
- Power BI
- Microsoft Flow
Create a canvas app

1. Connect to a data source

1. Go to https://web.powerapps.com and sign in with your organizational account.
2. In the left pane, select Apps.
3. Select Create an app.
4. Select your data source.
5. Under Connections, select OneDrive for Business.
6. Select Create and choose your data source.
7. Select Connect.

PowerApps generates the app by inspecting your data and matching it with PowerApps capabilities so that you get a working app as a starting point.

2. Explore the generated app

Your new three-screen app opens in PowerApps Studio. On the left, you’ll see the Screens pane. In the upper-right corner of the screens pane, select the thumbnail view. Select the thumbnail for each screen to view the controls on that screen.

Here is the main development window for PowerApps Studio, which you’ll learn more about in later units. Select Play in the upper-right corner to try out the app. You’ll see that it includes all the data from the list and provides a good default experience.

All apps generated from data have the same set of screens that you can view from the Thumbnail pane:

- **Browse screen**: This screen appears by default. In it, you can browse, sort, filter, and refresh the data from the data source. In the browse screen, you add items to the data source by selecting the plus sign (+).

- **Details screen**: Select an item in the browse screen to open the details screen, which shows all details about an item. In this screen, you can open an item for editing or delete it.

- **Edit/create screen**: In this screen, you edit an existing item, or you create one.

Create accessible canvas apps

Create accessible canvas apps in PowerApps, Use the Accessibility Checker to help review potential accessibility issues in your app.

*Note: Generated apps are always based on a single list or table, but you can add more data to the app later.*
Building your application

Install the app on your device
You’ll want to install your app on your phone to see how it looks there.

1. Download PowerApps Mobile from the app store for the platform that you want to use.
2. Sign in by using your user name and password.
3. On your phone or tablet, run the flooring-estimates app in PowerApps Mobile. Otherwise, run the app in a browser.

In just a few minutes, you learned how to connect to a data source and generate an app. You also got acquainted with PowerApps Studio and the three screens in a generated app. In later modules, you’ll learn how to customize generated apps.

Customize your app
With the foundation of your canvas app established, you can customize your apps functionality and design.

• Customize a gallery
• Customize a form
• Customize a card
• Configure an app control
• Configure app functionality
Building your application

Use App checker to fix errors and make accessible apps
The App checker is helps provide a clear list of formula issues in your app and items to fix to make your app accessible. The App checker is an area that the PowerApps team will continue to invest in, and build on in order help to make debugging, performance and best practice decisions an easier and more guided experience.

Sharing canvas apps
After you build a canvas app that addresses a business need, you’ll want to share your app. First, specify which users in your organization can run the app and which can modify and even re-share it. Then, specify each user by name or specify a security group in Azure Active Directory. If everyone would benefit from your app, specify that your entire organization can run it.

Manage a canvas app
Edit any canvas app that you built, that you own, or for which you have “Can edit” permissions. You can edit an app in PowerApps Studio. If you try to edit an app that’s open for editing elsewhere, a message tells you whether you already have it open or another user does. Learn how to edit or delete an app, restore a previous version, name and tile, change the screen size and orientation, or customize a SharePoint list form.

Create a model-driven app

Building blocks of model-driven apps
A model-driven app consists of several components that you select by using the App Designer. The components and component properties become the meta data.

- **Data**: The data components determine what data the app will be based upon.
- **User interface**: The user interface components determine how users will interact with the app.
- **Logic**: The logic components determine what business processes, rules, and automation the app will have. Microsoft PowerApps makers use a designer that’s specific to the type of process or rule.
- **Visualization**: The visualization components determine what type of data and reporting the app will show.
- **Advanced model-driven apps**: Solution Explorer is used to make advanced model-driven apps. By using the navigation pane on the left side of the tool, you can navigate a hierarchy that consists of all app components.
Building your application

Design phases
1. Model your business data
Model-driven design uses meta data-driven architecture so that designers can customize apps without writing code. To model business data, you determine what data the app will need and how that data will relate to other data. Meta data means *data about data* and it defines the structure of the data stored in Common Data Service for Apps.

2. Define your business processes
Defining and enforcing consistent business processes is a key aspect of model-driven app design. Consistent processes help ensure that your app users can focus on their work and not worry about having to remember to perform a set of manual steps. Processes can be simple or complex, and they often change over time.

3. Build the app
After modeling data and defining processes, you build your app by selecting and setting up the components you need in the App Designer.

Create your model-driven app
Create a model-driven app by using one of the standard entities that’s available in your Microsoft PowerApps environment.

1. Create a model-driven app
1. Sign in to PowerApps by using your organizational account.
2. Select the environment you want or go to the PowerApps admin center to create a new one.
3. On the Home page, select the Start from blank option for a model-driven app.*

*Note: If the Model-driven design mode isn’t available, you might need to create an environment.
Building your application

4. In the left pane, select **Apps**, then select **Create an app**.

5. On the **Create a New App** page, enter a name and description for the app.

6. Select **Done**. Your new app appears in the App Designer, and you can now add components to it.

2. Add components to your app

   You add components to your app by using the App Designer.

   1. Select the **Open the Site Map Designer** arrow to open the site map designer.

   2. In the site map designer, select **New Subarea**, and then, in the right pane on the **Properties** tab, select the following properties:
      - **Type**: Entity
      - **Entity**: Account

   3. Select Save And Close.

   4. In the App Designer, select **Forms**, and then, in the right pane under **Main Forms**, select the Account form.

   5. In the App Designer, select **Views**, then select the following properties:
      - **Active Accounts**
      - **All Accounts**
      - **My Active Accounts**

   6. In the App Designer, select **Charts**, then select the **Accounts by Industry** chart.

   7. On the App Designer toolbar, select **Save**.

3. Publish your app

   On the App Designer toolbar, select Publish. After you publish the app, it’s ready for you to run or share with others.

Customize your apps

You can tailor a model-driven app to more closely fit your organization’s industry, nomenclature, and unique business processes. You can implement many customizations without writing any code.

- **Create an app using the app designer**
- **Create and design forms**
- **Create or edit views**
- **Create or edit a system chart**
- **Create or edit dashboards**
- **Add security roles**
- **Add business logic**
Use App checker to fix errors and make accessible apps

The App checker helps provide a clear list of formula issues in your app and items to fix to make your app accessible. The App checker is an area that the PowerApps team will continue to invest in, and build on in order help to make debugging, performance and best practice decisions an easier and more guided experience.

Sharing model-driven apps

PowerApps apps use role-based security for sharing. To share a model driven app, you must create a custom security role, assign users to the custom security role, and assign the security role to an app. Then you can share the app.

Managing the properties of a model-driven app

App properties define important details about the app, like its title or URL. You define app properties when you create an app. You can change and manage those properties in the app designer.

Properties include: Name, Description, Icon, Unique Name, Client, App URL Suffix, App welcome page, and Enable Mobile Offline.

3 Build with Power BI and Microsoft Flow

Power BI

Browse the Power BI documentation to learn more about building powerful dashboards and beautiful data visualizations.

Microsoft Flow

Browse the Microsoft Flow documentation to learn more about building Connectors and automating workstreams.
Publish your PowerApp to AppSource

Step 1  
Create a managed solution package
  • Export solution to Package Deployer
  • Create a package using Package Deployer
  • Upload package to Azure

Step 2  
Validate your code
  • QA your code
  • Validate your code using the On-Demand Code Analysis tool

Step 3  
Create a new offer
  • Create a new offer in the Cloud Partner Portal

Step 4  
Submit for publication
  • Click Publish to submit your app for approval
  • Steps to publication

Step 5  
Publish your solution
  • View and publish your solution
  • Monitor performance and collect leads

Step 6  
Market and sell
  • Leverage Microsoft support to market and sell your app
Package and publish your solution

Once you’ve created your application and are ready to share it with the world, there are just a handful of steps you must complete to publish it on AppSource.

<table>
<thead>
<tr>
<th>Schema</th>
<th>User interface</th>
<th>Analytics</th>
<th>Process/Code</th>
<th>Templates</th>
<th>Security</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Entities</td>
<td>• Application ribbon</td>
<td>• Dashboards</td>
<td>• Processes</td>
<td>• Mail-merge</td>
<td>• Security roles</td>
</tr>
<tr>
<td>• Attributes</td>
<td>• SiteMap</td>
<td>• Reports</td>
<td>• Dialogs and workflows</td>
<td>• E-mail</td>
<td>• Field level</td>
</tr>
<tr>
<td>• Relationships</td>
<td>• Forms</td>
<td>• Visualizations</td>
<td>• Plug-ins</td>
<td>• Contract</td>
<td>security profiles</td>
</tr>
<tr>
<td>• Global option</td>
<td>• Entity ribbons</td>
<td></td>
<td>• Assemblies</td>
<td>• Article</td>
<td></td>
</tr>
<tr>
<td>sets</td>
<td>• Web resources</td>
<td></td>
<td>• Processing steps</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Solution component examples

1 Create a managed solution package

There are two types of solutions for Dynamics 365 Customer Engagement: managed and unmanaged. An unmanaged solution is one that is still under development or isn’t intended to be distributed; unmanaged solutions can still be edited. Once your unmanaged solution is ready to be distributed, you must export the unmanaged solution as a managed solution. A managed solution is a completed solution—with publishable code—that is intended to be distributed and installed by users.

Export solution to Package Deployer

When you are ready to publish, you must create an AppSource Package. Export your unmanaged solution as a managed solution, including customizations and meta data, to Package Deployer. Dynamics 365 provides you with a Visual Studio template for exporting packages that can be used with the Package Deployer tool.

For software that operates outside the application, use traditional methods to package and install your application, such as an installer program. If your application consists only of Dynamics 365 solution components, it can be imported directly into Dynamics 365. You won’t have to create an installer program. However, if your extensions include a combination of Dynamics 365 solution components and external components, you’ll need an installer.
Create a package using Package Deployer
Using Package Deployer, create a package (.zip) with your assets. The Package Deployer creates a solution package with the structure needed to submit to Microsoft for certification.

Creating a package involves creating a project using the template, adding your files to the project, updating the HTML files, specifying the configuration values for the package, and defining custom code for your package.

An AppSource package consists of:

- **Package file**: A package file used by Package Deployer to deploy your solutions and demo configuration data into multiple languages.
- **[Content_Types].xml**: File that provides MIME type information of the file type extensions included in the AppSource package. Typically, these are .config, .dll, .exe, .xml, and .zip file types, but you can add almost any file type that is supported by Windows.
- **Icon file**: An image file for the AppSource package icon; size should be 32x32 pixels. Valid image formats are PNG and JPG.
- **HTML file**: File containing your License terms.
- **Input.xml**: Files that describe the assets in your AppSource package.

Upload package to Azure
Once you have created your solution package, you must upload it to Azure. Before you upload, you should download and install the Microsoft Azure Storage Explorer, which enables you to manage the contents of your Azure storage account easily. The Microsoft Azure Storage Explorer can be downloaded at http://storageexplorer.com.

Storage account - blob, file, table, queue
Quickstart tutorial
Package and publish

2 Validate your code

QA your code
Before submitting your code for certification, it is always recommended to thoroughly QA your code, especially in the following areas:

- **Usage**: Improper usage of a particular API, pattern, or configuration
- **Design**: Design flaws in a customization
- **Performance**: Customization or pattern that may produce a negative effect on performance in areas such as memory management, CPU utilization, network traffic, or user experience
- **Security**: Potential vulnerabilities in a customization that could be exploited in a runtime environment
- **Upgrade Readiness**: Customization or pattern that may increase risk of having an unsuccessful version upgrade
- **Online Migration**: Customization or pattern that may increase risk of having an unsuccessful online migration
- **Maintainability**: Customization that unnecessarily increases the amount of developer effort required to make changes, the frequency of required changes, or the chance of introducing regressions
- **Supportability**: Customization or pattern that falls outside the boundaries of published supportability statements, including usage of removed APIs or implementation of forbidden techniques

Additionally, we recommend you follow these best practices when building your app, as it can expedite the certification process.

Validate your PowerApp
- Build model-driven apps of higher quality with Solution checker
- Build Canvas driven apps with App checker
3 Create an offer

Create a new offer in the Cloud Partner Portal

Once you resolved any issues identified by the On-Demand Code Analysis tool, you can move on to creating an offer in the Cloud Partner Portal.

Each app in AppSource corresponds to an “Offer” in the Cloud Partner Portal. To submit a new app to be published on AppSource, you must create a “New offer” in the portal. There are different offer types based on different types of apps. When creating your offer, you will be asked to provide information, including:

- **Offer Settings**: Offer ID, Publisher ID, Name
- **Technical Info**: Meta data (e.g., categories, regions, marketing artifacts, and Azure storage URL of the solution package)
- **Test Drive**: (Optional) Provide a test environment where users can use and explore a functional version of the app that has been preconfigured with demo data
- **Storefront Details**: Offer summary, Offer description, Industries, Categories, Terms, etc.
- **Contacts**: Engineering Contact, Support Contact

If you are an Azure partner already registered in the Cloud Partner Portal, you don’t need to register again; however, we recommend you submit the signup form to connect with a member of our Customer Engagement program management team.

About Test Drive

Are you passionate about building canvas apps in PowerApps? Do you want to share a canvas app with customers?

AppSource supports PowerApps Test Drive solutions as a way for you to share apps with customers and generate leads for your business.

Let customers test drive your canvas app on AppSource.

TEST DRIVE
4 Submit for publication

Click Publish to submit your app for approval

Once you have completed filling in the offer details, click Publish to begin the publishing approval process. Progress can be tracked on the offer page.

Steps to publication

- **Validate prerequisites**: Offer settings provided are validated (<15 min)
- **Test Drive Validate**: Microsoft validates the Test Drive can be deployed and be replicated (<2 hours)
- **Certify Package**: This process will encompass certifying your package for CRM deployment (~15 days) Precertification process can expedite this step.
- **Provision Package**: When complete, we will have deployed your package for use in the CRM Regions requested (~4 min)
- **Lead management validation and registration**: Microsoft validates and registers lead management details. (<15 min)
- **AppSource Packaging**: Offer is packaged to show up on AppSource (<1 hour)
- **Publisher signoff**: Offer is available to preview. Ensure that everything is correct before making your offer live.

![Image of publishing process](image_url)
Package and publish

5 Publish your solution

View and publish your solution
Once your app has been approved for publication on AppSource by Microsoft, you will receive a preview link to your offer, where you will be able to view your offer on AppSource and test as if were live. Once you are ready, you can set it to go live, at which time it will become publicly available on AppSource.

Monitor performance and collect leads
Within the offer page in the Cloud Partner Portal, you can view app performance. You will also receive leads from users who registered for Test Drive or Trial, requested to be contacted through the “Contact Me” form, or opted to share their information by selecting “Get it now.”.

6 Market and sell

Leverage Microsoft support to market and sell your app
As soon as your app is published, you can take advantage of Microsoft’s Go-to-market services, which will help you promote and sell your app. You may also be eligible to participate in other partner programs, such as the IP Co-Sell program. Learn more about our Go-to-market services at https://partner.microsoft.com/reach-customers/gtm.
ISV Cloud Embed Program
If you have built or intend to build a vertical/industry focused solution on top of Dynamics 365 (or horizontal solution on PowerApps) and publish on AppSource, the Microsoft ISV Cloud Embed program simplifies app development time, lowers costs, and helps your business grow. The program allows ISV partners to focus on continuous innovation and rapidly build business applications by extending Dynamics 365 applications or building on PowerApps; benefit from a growing community of Office 365 and Dynamics 365 customers through AppSource; and receive critical Go-To-Market and benefits culminating in co-sell support from one of the world's largest enterprise salesforces.

The requirements for the program are:
1. End-to-end solution built on Dynamics 365 or Power platform (PowerApps)
2. Solution published on AppSource
3. Become CSP Direct Bill partner through this link

It is important to note that you can only sell the Embedded SKUs with your IP as a Unified Solution; you cannot sell the Embedded SKUs separate from your IP and you will have to give one price point to the entire Unified Solution.

You can review the Program Overview, Licensing guide, and ISV Cloud Embed Handbook to familiarize yourself with the ISV Cloud Embed Program.
Additional resources
Additional resources

Glossary

**Azure:** Extensibility platform that allows ISVs to build, test, deploy, and manage applications within Microsoft PowerApps.

**Canvas App:** Microsoft PowerApps allows you to design and build a business app from a canvas without writing code in a traditional programming language such as C#. Design the app by dragging and dropping elements onto a canvas, just as you would design a slide in PowerPoint. Create Excel-like expressions for specifying logic and working with data. Build apps that integrate business data from a wide variety of Microsoft and third-party sources. Share your app so that users can run it in a browser or on a mobile device and embed your app so that users can run it in SharePoint, Power BI, or Teams. Building from a canvas gives you maximum flexibility over the appearance of the app.

**Connectors:** The elements that connect PowerApps and APIs.

**Common Data Model (CDM):** A standardized, modular, and extensible collection of data schemas that Microsoft published to help ISVs and users build, use, and analyze data. This collection of predefined schemas consists of entities, attributes, semantic metadata, and relationships. The CDM simplifies data management and app development by unifying data into a known form and applying structural and semantic consistency across multiple apps and deployments.

**Common Data Service (CDS):** The CDS makes it easier to bring your data together and quickly create powerful apps using a compliant and scalable data service and app platform that’s integrated into PowerApps.

**Microsoft Flow:** A business service that allows line of business specialists and IT pros to build automated workflows intuitively.

**Model-Driven App:** Model-driven app design is an approach that focuses on adding dashboards, forms, views, and charts to your apps. If you don’t need a custom design and your data is in Common Data Service (CDS) for Apps, you can automatically generate a model-driven app from your business data and processes. This type of app can model forms, views, and other components, and the default UI automatically adjusts to phones, laptops, and other devices. The emphasis is more on quickly viewing your business data and making decisions than on intricate app design.

**PowerApps:** A SaaS application platform that enables power users in line of business roles to easily build and deploy custom business apps.

**Power BI:** Self-service business intelligence capabilities, where end users can create reports and dashboards by themselves, without having to depend on IT staff or database administrators.
Additional resources

**Best practices and common use cases**

- **Community content**: Community apps gallery | Working with data | Design discussion
- **Real world solutions using PowerApps**: Video
- **UX/UI design tips**: Video | UX patterns | Managing fonts and colors
- **Customize list forms in SharePoint with PowerApps**: Blog post | Demo | Documentation
- **Access web apps migration to PowerApps**: Whitepaper
- **Approval workflows**: Blog post | Documentation | Guided learning
- **Using the on-premises gateway**: Overview | Installation and FAQ | Proxy configuration
- **Build an offline app**: Blog post
- **Creating dialogs in PowerApps**: Blog post | Video
- **Display a map using the Image control**: Step-by-step walkthrough | Video
- **Send an email from your PowerApps app**: Video
- **Connect to Microsoft Cognitive Services**: Video
- **Implement role-based security**: Blog post
- **Build a customized interactive calendar**: Step-by-step guide
- **Notify user that new data is available**: Step-by-step guide | Push notifications feature
- **Deep link into an app using URL parameters**: Step-by-step guide
- **Audit scenario solution**: Blog post from community member
- **Performance considerations when working with PowerApps**: Blog post

**Learning resources**

- **PowerApps latest feature updates**: Blog post | Release notes
- **Flow latest feature updates**: Blog post | Release notes
- **Browse presentations from the Ignite 2018 conference**: Blog post with curated links to relevant sessions
- **Browse Microsoft Business Applications Summit 2018 conference**: Blog post with curated list to relevant sessions
- **Browse How-to videos in the PowerApps video gallery | Flow Webinars and Videos gallery**
- **Get expert help from partners**: https://powerapps.microsoft.com/partners
- **SharePoint welcomes PowerApps and Flow**
- **PowerApps integration with SharePoint**: Ignite presentation
- **Introduction to Microsoft PowerApps for Access web apps developers**

**Stay connected**

- **Ways to collaborate with the PowerApps team**: Blog post
- **Stay updated**: Product team blog | Webinars
- **Community sites**: Forums | Community blog | Best practices and app gallery
- **Get support**: Blog post | Support page | Submit an idea

**Other tools**

- **Microsoft App in a Day Lab**
- **PowerApps and Microsoft Flow Learning Resources**
Getting Started Guides

Interest in building apps for Dynamics 365 for Finance and Operations and using Power platform? Check out our other Getting Started Guides.
Disclaimer

© 2019 Microsoft. All rights reserved.

This document is provided “as-is.” Information and views expressed in this document, including URL and other Internet Web site references, may change without notice. You bear the risk of using it. Some examples are for illustration only and are fictitious. No real association is intended or inferred.

This document does not provide you with any legal rights to any intellectual property in any Microsoft product. You may copy and use this document for your internal, reference purposes.