# Application Installation Guide

Version 39.0, Spring '17

# CONTENTS

# CHAPTER 1    What Is the AppExchange?

The AppExchange is a complete marketplace offering cloud-computing applications and consulting services. Before you can make your own app or service publicly available to potential customers, you must be a Salesforce partner.

# How Does AppExchange Work?

The creative power behind each AppExchange app is a Salesforce user. AppExchange gives users a place to create, publish, or install apps and extensions to Salesforce. Briefly, here is how Force.com AppExchange works.

**Create**

A Salesforce user initiates the process by creating a custom object, custom tab, or set of dashboards and reports that improves business. Customers can share their customizations with other Salesforce users as an app or extension.

**Publish**

A Salesforce customer can register and publish an app on the AppExchange. A customer can also create a demo of the functionality included in the app for others to view before installing.

**Install**

Salesforce admins can choose to install an app published to the AppExchange. Installing a published app from AppExchange is safe and simple.

- **Browse**—View the descriptions, reviews, and demos of an app from AppExchange. Choose an app that fits your needs.
- **Test Drive**—Review a fully functional demo of the app as a read-only user, and plan your implementation.
- **Install**—Add the app and all its components to your Salesforce environment.
- **Deploy**—Immediately give your users access to the app or customize it for a select group of users.

# Who Can Use the AppExchange?

Anyone can browse and test drive AppExchange listings. Salesforce administrators and users with the "Download AppExchange packages" permission can install AppExchange apps. To publish an app on the AppExchange, a user must have both "Create AppExchange packages" and "Upload AppExchange packages" permissions.

# CHAPTER 2  Before Installing a Package

Before you begin, review the apps and determine how they fit into your current implementation of Salesforce.

# Taking a Test Drive

A test drive is a new fully functional Developer Edition Salesforce environment that contains the app. If the app provider also provides a test drive, go to http://sites.force.com/appexchange to test drive the app before you install it.

# Verifying Requirements

Every application has its own requirements. To verify that an app is compatible with your organization, click the Spec tab and review the system requirements information.

# Determining if the App Supports Upgrades

Upgradeable apps are those that come from managed packages.

# CHAPTER 3    Installing Packages

To install a package from AppExchange, follow this procedure from the beginning. If you received an installation link, click the link and skip to the Enter the requested information step.

> ⚠ **Warning:** If the app is in a managed package, you cannot change some Salesforce components after you install them. For a list of editable components in a managed package, see the *ISVforce User Guide*. Managed packages also have a namespace prefix that is prepended to all uniquely named components, such as custom fields. For more information on namespace prefixes, see "Register a Namespace Prefix" in the Salesforce online help.

> 📝 **Note:** Only one person in your organization can install or upload a package at a time.

1. Click the **AppExchange** link in your organization or go directly to www.appexchange.com.

2. Browse the AppExchange site to find the application you want to install.

3. Click **Get It Now** from the application information page.

   If the listing does not have a **Get It Now** option, it may be a partner app that you can download and use outside of Salesforce. Contact the publisher for more information.

4. Select your environment type: production or sandbox (test).

5. Enter the requested information. For production environments, enter your username and password, read the user terms, select the `I have read and agree...` checkbox, and click **Install**.

   For sandbox environments, complete the information in the form provided and click **Submit**. This package is removed from your sandbox organization whenever you create a new sandbox copy.

6. If the package is password-protected, enter the password you received from the publisher.

7. Review the package items and click **Continue**.

   If you're installing an upgrade to a managed package, review the list of new items that the upgrade will add.

   > 📝 **Note:** Some package items, such as validation rules or record types, may not display in the Package Item list but are included in the package and install with the other items. If there are no items in the Package Items list, the package may contain only minor changes not recorded in the Package Items list.

8. Review the API access that package components have been granted to ensure they are acceptable, and click **Next** if they are. If they are not, click **Cancel**. For more information, see "About API and Dynamic Apex Access in Packages" in the Salesforce online help.

9. When installing unmanaged packages, you can select `Resolve Naming Conflicts Automatically`. When this is selected, Salesforce changes the name of the component in the package being installed.

10. In Enterprise, Unlimited, Performance, and Developer Editions, choose a security option, and click **Next**:

    **Install for Admins Only**

    Specifies the following settings on the installing administrator's profile and any profile with the "Customize Application" permission.

    - Object permissions—"Read," "Create," "Edit," "Delete," "View All," and "Modify All" enabled
    - Field-level security—set to visible and editable for all fields
    - Apex classes—enabled
    - Visualforce pages—enabled
    - App settings—enabled
    - Tab settings—determined by the package creator
    - Page layout settings—determined by the package creator
    - Record Type settings—determined by the package creator

    After installation, if you have Enterprise, Performance, Unlimited, or Developer Edition, set the appropriate user and object permissions on custom profiles as needed.

    **Install for All Users**

    Specifies the following settings on all internal custom profiles.

    - Object permissions—"Read," "Create," "Edit," and "Delete" enabled
    - Field-level security—set to visible and editable for all fields
    - Apex classes—enabled
    - Visualforce pages—enabled
    - App settings—enabled
    - Tab settings—determined by the package creator
    - Page layout settings—determined by the package creator
    - Record Type settings—determined by the package creator

    Note: The Customer Portal User, Customer Portal Manager, High Volume Customer Portal, Authenticated Website, Partner User, and standard profiles receive no access.

    **Install for Specific Profiles...**

    Enables you to choose the usage access for all custom profiles in your organization. You can set each profile to have full access or no access for the new package and all its components.

    - Full Access—Specifies the following settings for each profile.
        - Object permissions—"Read," "Create," "Edit," "Delete," "View All," and "Modify All" enabled
        - Field-level security—set to visible and editable for all fields
        - Apex classes—enabled
        - Visualforce pages—enabled

6

       – App settings—enabled

       – Tab settings—determined by the package creator

       – Page layout settings—determined by the package creator

       – Record Type settings—determined by the package creator

- No Access—Specifies the same settings as Full Access, *except* all object permissions are disabled.

You might see other options if the publisher has included settings for custom profiles. You can incorporate the settings of the publisher's custom profiles into your profiles without affecting your settings. Choose the name of the profile settings in the drop-down list next to the profile that you need to apply them to. The current settings in that profile remain intact.

Alternatively, click **Set All** next to an access level to give this setting to all user profiles.

> 📝 Note: Depending on the type of installation, you might not see this option. For example, in Group and Professional Editions, or if the package doesn't contain a custom object, Salesforce skips this option, which gives all users full access.

**11.** In Enterprise, Unlimited, Performance, and Developer Editions, if you chose **Select security settings**, select the level of access to give users in each profile and click **Next**. Standard options are:

- Full Access—Specifies the following settings for each profile.

       – Object permissions—"Read," "Create," "Edit," "Delete," "View All," and "Modify All" enabled

       – Field-level security—set to visible and editable for all fields

       – Apex classes—enabled

       – Visualforce pages—enabled

       – App settings—enabled

       – Tab settings—determined by the package creator

       – Page layout settings—determined by the package creator

       – Record Type settings—determined by the package creator

- No Access—Specifies the same settings as Full Access, *except* all object permissions are disabled.

You might see other options if the publisher has included settings for custom profiles. You can incorporate the settings of the publisher's custom profiles into your profiles without affecting your settings. Choose the name of the profile settings in the drop-down list next to the profile that you need to apply them to. The current settings in that profile remain intact.

Alternatively, click **Set All** next to an access level to give this setting to all user profiles.

**12.** Click **Install** to install all the components in the package you have selected. See Configuring Installed Packages in the Salesforce online help to view the standard settings each type of component will have after you install.

> 📝 Note: During installation, Salesforce automatically resolves any references to package components from the source organization. For example, if a custom link in the package refers to a document, the ID for the document will be different once you install it so Salesforce automatically alters the installed custom link to refer to the new ID in your organization. Salesforce also checks and verifies any dependencies. An installer's organization must meet all dependency requirements listed on the Show Dependencies page or else the installation will fail. For example, the installer's organization must have divisions enabled to install a package

that references divisions. See Understanding Dependencies in the Salesforce online help for more information on package dependencies.

When you install a component that contains an Apex script, all unit tests for your organization are run, including the unit tests contained in the new package. If a unit test relies on a component that is initially installed as inactive, such as a workflow rule, this unit test may fail. You can select to install regardless of unit test failures.

For reasons Salesforce could prevent an install, see "Why did my installation or upgrade fail?" in the Salesforce online help.

**13.** Click **OK**.

**14.** Configure your package. See "Configuring Installed Packages" in the Salesforce online help.

**15.** If the package includes permission sets, assign the included permission sets to your users who need them. In managed packages, you can't make changes to permission sets that are included in the package, but subsequent upgrades happen automatically. If you clone a permission set that comes with a managed package or create your own, you can make changes to the permission set, but subsequent upgrades won't affect it.

**16.** If you are reinstalling a package, see Importing Package Data in the Salesforce online help to reimport the package data using the export file you received after uninstalling.

**17.** If you installed a managed package, click **Manage Licenses**, to assign licenses to users. See Managing Licenses for Installed Packages in the Salesforce online help.

# CHAPTER 4   App Rollout

In this chapter ...

- Review Best Practices for Use

Before you deploy the app to your users, notify them of the new functionality and educate them on how to use it. Then make it available to the users that should have permission to use it.

# Review Best Practices for Use

**Review the Splash Page**

The developer of an app can include a splash page that users can view when they click on a custom tab. Splash pages explain how the custom object fits in a Salesforce environment and can contain training or support links. Instruct users to review the splash pages for each custom tab. Show them how to use the **Don't show me this again** link. If the publisher did not provide splash pages, create your own.

**Preview the App for Select Users**

Preview the app with key users in your organization. Make any suggested changes from the feedback you receive.

**Announce the New App**

Send an announcement to your users one week before deploying the new app. Let them know when the rollout is and how the new app benefits them. Send another announcement on the day of the deployment telling users to look for the new app.

# CHAPTER 5    After Installing a Package

After installing an app from Force.com AppExchange, manage it like you manage other standard apps such as Sales or Service.

11

# Manage Installed Packages

Manage packages installed in your Salesforce org, including assigning licenses to users, uninstalling packages, and exporting package data.

> 📝 **Note:** Salesforce only lists license information for managed packages. For unmanaged packages, the license-related fields, such as **Allowed Licenses**, **Used Licenses**, and **Expiration Date**, displays the value "N/A."

Using this list, you can:

- Click **Uninstall** to remove the package and all its components from your Salesforce organization.
- Click **Manage Licenses** to assign available licenses to users in your organization. You can't assign licenses in Lightning Experience. If you need to assign a license, switch to Salesforce Classic.

  > 📝 **Note:** If you purchased a site license or if the managed package is not licensed, Salesforce assigns licenses to all your users and you can't manage licenses. Your users can use the package as long as they have the appropriate permissions.

- Click **Configure** if the publisher has included a link to an external website with information about configuring the package.
- Click the package name to view details about this package.
- View the publisher of the package.
- View the status of the licenses for this package. Available values include:
    - Trial
    - Active
    - Suspended
    - Expired
    - Free

  This field is only displayed if the package is managed and licensed.

- Track the number of licenses available (`Allowed Licenses`) and the number of licenses that are assigned to users (`Used Licenses`).
- View the date your licenses for this package are scheduled to expire.
- View the date your licenses were installed.
- View the number of custom apps, tabs, and objects this package contains.
- See whether the custom apps, tabs, and objects count toward your organization's limits. If they do, the box in the `Limits` column is checked.

> 📝 **Note:** If you have not installed a licensed managed package, the `Publisher`, `Status`, `Allowed Licenses`, `Used Licenses`, and `Expiration Date` fields do not appear.

After an uninstall, Salesforce automatically creates an export file containing the package data, as well as any associated notes and attachments. When the uninstall is complete, Salesforce sends an email containing a link to the export file to the user performing the uninstall. The export file and related notes and attachments are listed below the list of installed packages. We recommend storing the file elsewhere because it's only available for a limited period of time after the uninstall completes. Using this list, you can:

- Click **Download** to open or store the export file.
- Click **Del** to delete the export file.

EDITIONS

EDITIONS

Available in: both Salesforce Classic and Lightning Experience

Available in: **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions

USER PERMISSIONS

To uninstall packages:
- "Download AppExchange Packages"

To assign licenses for a managed package:
- "Manage Package Licenses"

To download or delete the export file for an uninstalled package:
- "Download AppExchange Packages"

**Expired Managed Packages and Sharing Rules**

> If a criteria-based sharing rule references a field from a licensed managed package whose license has expired, `(expired)` is appended to the label of the field. The field label is displayed in the field drop-down list on the rule's definition page in Setup. Criteria-based sharing rules that reference expired fields aren't recalculated, and new records aren't shared based on those rules. However, the sharing of existing records prior to the package's expiration is preserved.

# Uninstalling a Package

You can remove any installed package, including all of its components and all data in the package. Additionally, any custom fields, links, or anything else you added to the custom app after installation are also removed.

To remove a package:

1. From Setup, enter `Installed` in the `Quick Find` box, then select **Installed Packages**.

2. Click **Uninstall** next to the package that you want to remove.

3. Select `Yes, I want to uninstall...` and click **Uninstall**.

4. After an uninstall, Salesforce automatically creates an export file containing the package data, as well as any associated notes and attachments. When the uninstall is complete, Salesforce sends an email containing a link to the export file to the user performing the uninstall. The export file and related notes and attachments are listed below the list of installed packages. We recommend storing the file elsewhere because it's only available for a limited period of time after the uninstall completes.

> 💡 **Tip:** If you reinstall the package later and want to reimport the package data, see Importing Package Data.

| EDITIONS |
| --- |
| Available in: both Salesforce Classic and Lightning Experience |
| Available in: **Group**, **Professional**, **Enterprise**, **Performance**, **Unlimited**, and **Developer** Editions |

| USER PERMISSIONS |
| --- |
| To uninstall packages: |
| • "Download AppExchange Packages" |

## Notes on Uninstalling Packages

- If you're uninstalling a package that includes a custom object, all components on that custom object are also deleted. This includes custom fields, validation rules, s-controls, custom buttons and links, as well as workflow rules and approval processes.

- You can't uninstall a package whenever any component in the package is referenced by a component that will not get included in the uninstall. For example:

  - When an installed package includes any component on a standard object that another component references, Salesforce prevents you from uninstalling the package. This means that you can install a package that includes a custom user field and build a workflow rule that gets triggered when the value of that field is a specific value. Uninstalling the package would prevent your workflow from working.

  - When you have installed two unrelated packages that each include a custom object and one custom object component references a component in the other, Salesforce prevents you from uninstalling the package. This means that you can install an expense report app that includes a custom user field and create a validation rule on another installed custom object that references that custom user field. However, uninstalling the expense report app prevents the validation rule from working.

  - When an installed folder contains components you added after installation, Salesforce prevents you from uninstalling the package.

  - When an installed letterhead is used for an email template you added after installation, Salesforce prevents you from uninstalling the package.

- You can't uninstall a package if a field added by the package is being updated by a background job, such as an update to a roll-up summary field. Wait until the background job finishes, and try again.

- Uninstall export files contain custom app data for your package, excluding some components, such as documents and formula field values.

# CHAPTER 6  FAQ

Review the following list of answers to frequently asked questions about installing apps from Force.com AppExchange.

# Can I uninstall packages that I installed from AppExchange?

Yes. All your installed packages are listed in the Installed Packages page. You can remove any package by clicking the **Uninstall** link next to the package name.

# Who can use AppExchange?

Anyone can browse and test drive AppExchange listings. Salesforce administrators and users with the "Download AppExchange packages" permission can install AppExchange apps. To publish an app on the AppExchange, a user must have both "Create AppExchange packages" and "Upload AppExchange packages" permissions.

# Why did my installation or upgrade fail?

An installation can fail for several reasons:

- The package includes custom objects that will cause your organization to exceed its limit of custom objects.
- The package includes custom tabs that will cause your organization to exceed its limit of custom tabs.
- The developer of the package has uploaded a more recent version of the package and has deprecated the version associated with this installation URL. Contact the publisher of the package to get the most recent installation URL.
- You're trying to install an extension to a package, and you don't have the base package installed.
- The package requires that certain components are enabled in your organization, or that required features are enabled in your edition.
- The package contains Apex code and you are not authorized to run Apex in your organization.
- The package you're installing has a failing Apex test.

# Can I customize AppExchange packages?

Yes, all packages are customizable. However, to ensure compatibility with future versions, some aspects of managed packages can't be changed.

For a list of components that are editable in a managed package, see ISVforce Guide.

# Who can use AppExchange packages?

If you use an Enterprise, Unlimited, Performance, or Developer Edition organization, you can choose which user profiles have access to the package as part of the installation process. Packages installed in Professional and Group Edition organizations are installed with "Full Access" to all user profiles. However, regardless of Edition, all custom objects are installed in "In Development" mode which hides them from all standard users. Users must have the "Customize Application" permission to view custom objects in "In Development" mode. When you are ready to roll out the package to other users, change the custom object status to "Deployed."

# How can I upgrade an installed package?

Managed packages are completely upgradeable. Before installing a package, contact the publisher to determine if it's managed.

# How secure are the components I install?

Salesforce performs periodic security reviews of all publicly listed applications on AppExchange. When installing third party applications with access to data, these applications may have access to other data within the organization where the package was installed. Private listings do not go through a security review and administrators should inspect the application carefully before determining whether it should be installed within their organization.

# What happens to my namespace prefix when I install a package?

A namespace prefix is a globally unique identifier that you can request if you plan to create a managed package. All the components from a managed package that you install from another developer contain the developer's namespace prefix in your organization. Unmanaged packages can have a namespace prefix while they're developed in an org that contains a managed package. This namespace isn't used outside of the development (publisher) org. If an unmanaged package is installed in an org that has no namespace, then the unmanaged components have no namespace in the subscriber org. If an unmanaged package is installed in an org that has a namespace, then the components get the namespace of the subscriber org.

# Can I reinstall an AppExchange package after uninstalling it?

Yes. You can reinstall a package in the same manner that you installed it.

# When I install a package that's listed on the AppExchange, do custom objects, tabs, and apps in that package count against the limits of my Salesforce Edition?

No. If you install a package from the AppExchange, its custom objects, tabs, and apps don't count against the limits of your Salesforce edition. However, if the package uses other types of custom components, such as custom fields, they count against the relevant limits of your Salesforce edition.

Note: These rules apply only to managed packages that are listed on the AppExchange. If you install an unmanaged package or a managed package that's not publicly listed on the AppExchange, its custom objects, tabs, and apps count against the limits of your Salesforce edition.

# CHAPTER 7   Glossary

The following terms and definitions describe key application and packaging concepts and capabilities:

**App**

Short for "application." A collection of components such as tabs, reports, dashboards, and Visualforce pages that address a specific business need. Salesforce provides standard apps such as Sales and Service. You can customize the standard apps to match the way you work. In addition, you can package an app and upload it to the AppExchange along with related components such as custom fields, custom tabs, and custom objects. Then, you can make the app available to other Salesforce users from the AppExchange.

**AppExchange**

The AppExchange is a sharing interface from Salesforce that allows you to browse and share apps and services for the Force.com platform.

**Beta, Managed Package**

In the context of managed packages, a beta managed package is an early version of a managed package distributed to a sampling of your intended audience to test it.

**Deploy**

To move functionality from an inactive state to active. For example, when developing new features in the Salesforce user interface, you must select the "Deployed" option to make the functionality visible to other users.

The process by which an application or other functionality is moved from development to production.

To move metadata components from a local file system to a Salesforce organization.

For installed apps, deployment makes any custom objects in the app available to users in your organization. Before a custom object is deployed, it is only available to administrators and any users with the "Customize Application" permission.

**License Management Application (LMA)**

A free AppExchange app that allows you to track sales leads and accounts for every user who downloads your managed package (app) from the AppExchange.

**License Management Organization (LMO)**

The Salesforce organization that you use to track all the Salesforce users who install your package. A license management organization must have the License Management Application (LMA) installed. It automatically receives notification every time your package is installed or uninstalled so that you can easily notify users of upgrades. You can specify any Enterprise, Unlimited, Performance, or Developer Edition organization as your license management organization. For more information, go to http://www.salesforce.com/docs/en/lma/index.htm.

**Major Release**

A significant release of a package. During these releases, the major and minor numbers of a package version increase to any chosen value.

**Managed Package**

A collection of application components that is posted as a unit on the AppExchange and associated with a namespace and possibly a License Management Organization. To support upgrades, a package must be managed. An organization can create a single managed package that can be downloaded and installed by many different organizations. Managed packages differ from unmanaged packages by having some locked components, allowing the managed package to be upgraded later. Unmanaged packages do not include locked components and cannot be upgraded. In addition, managed packages obfuscate certain components (like Apex) on subscribing organizations to protect the intellectual property of the developer.

**Managed Package Extension**

Any package, component, or set of components that adds to the functionality of a managed package. You cannot install an extension before installing its managed package.

**Namespace Prefix**

In a packaging context, a namespace prefix is a one to 15-character alphanumeric identifier that distinguishes your package and its contents from packages of other developers on AppExchange. Namespace prefixes are case-insensitive. For example, ABC and abc are not recognized as unique. Your namespace prefix must be globally unique across all Salesforce organizations. It keeps your managed package under your control exclusively.

**Package**

A group of Force.com components and applications that are made available to other organizations through the AppExchange. You use packages to bundle an app along with any related components so that you can upload them to AppExchange together.

**Package Dependency**

This is created when one component references another component, permission, or preference that is required for the component to be valid. Components can include but are not limited to:

- Standard or custom fields
- Standard or custom objects
- Visualforce pages
- Apex code

Permissions and preferences can include but are not limited to:

- Divisions
- Multicurrency
- Record types

**Package Installation**

Installation incorporates the contents of a package into your Salesforce organization. A package on the AppExchange can include an app, a component, or a combination of the two. After you install a package, you may need to deploy components in the package to make it generally available to the users in your organization.

**Package Version**

A package version is a number that identifies the set of components uploaded in a package. The version number has the format *majorNumber.minorNumber.patchNumber* (for example,

2.1.3). The major and minor numbers increase to a chosen value during every major release. The *patchNumber* is generated and updated only for a patch release.

Unmanaged packages are not upgradeable, so each package version is simply a set of components for distribution. A package version has more significance for managed packages. Packages can exhibit different behavior for different versions. Publishers can use package versions to evolve the components in their managed packages gracefully by releasing subsequent package versions without breaking existing customer integrations using the package. See also Patch and Patch Development Organization.

**Patch**

A patch enables a developer to change the functionality of existing components in a managed package, while ensuring subscribing organizations that there are no visible behavior changes to the package. For example, you can add new variables or change the body of an Apex class, but you may not add, deprecate, or remove any of its methods. Patches are tracked by a *patchNumber* appended to every package version. See also Patch Development Organization and Package Version.

**Patch Development Organization**

The organization where patch versions are developed, maintained, and uploaded. Patch development organizations are created automatically for a developer organization when they request to create a patch. See also Patch and Package Version.

**Patch Release**

A minor upgrade to a managed package. During these releases, the patch number of a package version increments.

**Publisher**

The publisher of an AppExchange listing is the Salesforce user or organization that published the listing.

**Push Upgrade**

A method of delivering updates that sends upgrades of an installed managed package to all organizations that have installed the package.

**Subscriber**

The subscriber of a package is a Salesforce user with an installed package in their Salesforce organization.

**Test Drive**

A test drive is a fully functional Salesforce organization that contains an app and any sample records added by the publisher for a particular package. It allows users on AppExchange to experience an app as a read-only user using a familiar Salesforce interface.

**Unmanaged Package**

A package that cannot be upgraded or controlled by its developer.

**Upgrading**

Upgrading a package is the process of installing a newer version. Salesforce supports upgrades for managed packages that are not beta.

**Uploading**

Uploading a package in Salesforce provides an installation URL so other users can install it. Uploading also makes your packaged available to be published on AppExchange.

# INDEX