

The Big Book on Salesforce AppExchange

How to build and distribute applications on Salesforce's AppExchange

AppExchange Applications- First Things First

Overview

Let us demystify AppExchange by defining some basic terms, understanding the power of the tools and knowing how to handle the issues that real-life customers face with AppExchange. Just following a simple game plan and noting some critical things can make you sail through this world of AppExchange development and listing.

Brand Assets' Specifications

To demonstrate a high level of professionalism and impact your brand awareness positively, the application that you plan to create should be uniquely identifiable. For the same, these assets are highly recommended.

📁 **TILE:** shown in the top 6 search results - Tile Specs: 280x205 pixels, PNG only

- Try to communicate your value proposition in less than 5 words
- Use your logo to express your brand
- Color and gradient can help your app stand out
- Keep the text simple and minimal. Need not reveal all information in the tile.
- You might want to take help in designing the tile. This, in most cases, is the first thing that customers see.

📁 **LOGO:** Upload your listing's logos. Your logos show up in many places on the AppExchange, including on your listing, in search results, and on the home page. Check out our guidelines for creating a great logo. If your listing is private, if you do not upload listing logos, AppExchange displays default logos. To make your listing public, you must upload small, tile, and high-resolution logos, as described below.

- **Small Logo** - Upload a PNG file up to 100 KB. The logo must be exactly 60 × 60 pixels.
- **Tile Logo** - Upload a PNG file up to 300 KB. The logo must be exactly 280 × 205 pixels
- **High Resolution Logo** - Upload a high-resolution PNG file, with a transparent background, up to 10 MB. This logo is for marketing purposes only and is not used on the site

Effects like shadows are automatically applied, so you don't need to include them in your logo.

📁 **ICON:** shown in search results & on listing - Icon Specs: 60x60 pixels, PNG only

When searching listings on the AppExchange, potential customers can quickly identify features of your app or service with the icon image.

Icon description

	Signifies that your app is for sale (versus free).
	Signifies that you provide support for your app.
	Signifies that this app is a Native app.
	Identifies your salesforce.com partnership status as a registered consulting partner.

	Identifies your salesforce.com partnership status as a select consulting partner.
	Identifies your salesforce.com partnership status as a premier consulting partner.
	Identifies the number of Salesforce certified individuals in your organization.
	Identifies the number of completed projects your consulting company has registered with salesforce.com. On the listing page, the percentage of satisfied customers is shown.

- ◇ **BANNER:** shown on listing only & the biggest space to represent your brand - Banner Specs: 1200x300 pixels, PNG file upto 1 MB only. This banner is shown on your listing, summary page.
- ◇ **SCREENSHOT:** shown on listing - Screenshot Specs: 750x500 pixel jpg or png no larger than 1MB
- ◇ **VIDEO:** Provide a video about your listing. You have two options.
 - Specify a URL to view a video in a new page. This takes the customer away from your listing page, but enables you to generate leads.
 - Embed a YouTube video in the same page. This provides an integrated experience for the customer, but doesn't enable you to generate leads. You will need to specify a caption for the video.
- ◇ **IMAGES:** Upload up to eight images of your listing. There is no restriction on the size of the image. Files must be PNG and up to 1 MB.
- ◇ **RESOURCES:** Upload up to six whitepapers, testimonials, case studies, data sheets, webinars, customization guides, or other resources that can be hosted on the AppExchange or elsewhere. Resources can be PDF files up to 10 MB.

Basic of AppExchange Applications

General information

AppExchange is Salesforce's on-demand platform, a suite of development tools for customizing, building, integrating, and installing business applications. Tech savvy folks can surely code for the feeling of satisfaction they get out of it. However, for building applications you can use 'point and click' and quickly develop new applications in hours or days, completely integrated with your existing CRM.

To be able to develop, test, and sell your offering, you need to understand the process well. Failing the Salesforce Review can take your plans behind your target schedule.

Before we begin learning how to chart this path, knowing the meanings of a few basic terms will definitely help.

- ◇ **Environment Hub:** This lets you connect, create, view, and log in to Salesforce orgs from one location. You can connect existing Salesforce orgs to the Environment Hub, allowing you to manage all your development, test, and trial environments from one location. When you connect an org to the hub, related orgs are automatically discovered.
- ◇ **Orgs:** Force.com platform provides environments/ organizations/ orgs, which let you instantly start developing and testing your cloud computing application. As an AppExchange Partner, you have orgs for development, testing, staging and packaging, demos, marketing, selling, and distributing your app.
- ◇ **Partner Program:** The Partner Program gives you access to the Partner Community, where you can get help and training information, log cases for support issues, and collaborate with other partners. You also get access to the Environment Hub, which lets you create and manage new test and development orgs.

- ◇ **Standard objects** are tables that contain data on any standard tab, like an account or a contact. your created.
- ◇ **Custom objects** are tables you create to store other data that's typically outside the normal realm of the data in standard objects, like expenses, job postings, account plans, and features.
- ◇ **Custom tabs** are tabs where you can centralize information so that your users can be more effective. These tabs look just like the standard salesforce.com tabs.
- ◇ **Relationships** connect objects with each other. For example, accounts are linked with contacts through relationships.
- ◇ **Lookup fields** are fields on a record that look up other object records. For example, Account Name is a lookup field on an opportunity record that looks up accounts.
- ◇ **Picklist field** values for custom fields can be added, edited, or deleted by subscribers. Picklist values can be added or deleted by the developer. During a package upgrade, no new picklist values are installed into the subscriber 's organization for existing fields. Any picklist values deleted by the developer are still available in the subscriber 's organization.
- ◇ **Related lists** are lists on a record detail page of other object records that have been linked to the record through a lookup field
- ◇ **Visualforce** is the easiest way to model the design of your app after the Salesforce user interface look and feel. Apps that resemble the Salesforce user interface's look and feel are instantly more familiar to users and easy to use. When you use a standard controller with a Visualforce page, your new page takes on the style of the associated object 's standard tab in Salesforce.

Pricing

What Salesforce charges:

There is a one-time upfront fee, and a small subsequent annual fee for this process. The initial Security Review fee is \$2,700 USD for each paid app submitted (no fee for free apps). This includes the annual listing fee of \$150 USD for the first year. If your package only contains Lightning Components, you are eligible for a discount of \$1900, which means your security review fee will only be \$800. For up to date information on pricing, check FAQ section of the link [here](#).

What You plan to charge:

You can specify if your application is free or paid, and if you offer discounts for non-profits/NGOs. Pricing is ultimately at your discretion but the typical pricing models that the partners follow are either a \$/user/month or ELA model.

You can also sign up for Checkout, a service that lets your customers pay for your applications directly on the AppExchange. You need to specify how you want to collect payment information: before customers install your app or after they start a free trial.

◇ Other parameters that can be set are:

- Charge: price per unit
- Units: unit on which pricing depends
- Frequency: billing frequency
- Free trial: number of days a free trial is valid

- ◇ **Partner Developer Edition:** Also known as your dev org, this is where you develop your offering and eventually package it for distribution.
- ◇ **Test Organization:** Also known as your test org, this is where you install and test your offering.
- ◇ **Partner Business Organization:** This is where you license and manage your offering
- ◇ **Sandbox:** Is a replica of your customer's org that allows them to develop, test, or install apps, and verify the changes they want to commit. If you have a sandbox in an org and want to install your app in it, you must replace the initial portion of the Installation URL with <http://test.salesforce.com>.

◇ **Salesforce Editions:** Salesforce CRM is offered in five tiers, or editions:

- Group Edition (GE)*
- Professional Edition (PE)
- Enterprise Edition (EE)
- Unlimited Edition (UE)
- Performance Edition (PXE)*

EE/UE/PXE support Force.com platform licenses in addition to Salesforce CRM licenses. For GE/PE organizations, only existing Salesforce CRM users can use your app. Supporting multiple editions provides the opportunity to release richer versions of your app. To support multiple editions, the first approach is use of extension packages and the second is leveraging Dynamic Apex, dynamic SOQL, and dynamic DML (to create one managed package for all editions you plan to support).

◇ **Tab:** Container for things you want to display on the same page, such as a chart, a table, or the Visualforce page you created.

◇ **App:** Container for tabs that appear next to each other. When you create an app, it's available in the app picker in the upper right-hand corner of the screen.

◇ **App:** is primarily built on the Force.com platform except for callouts to the following providers: Google, Facebook, or Amazon. Native apps provide many of the same benefits as Salesforce CRM, which is why they are popular with customers.

◇ **Package:** Container for things you upload to the AppExchange, for distributing custom functionality between Salesforce orgs. Usually a package contains an app your customers can install in their org, but you can also upload packages that extend existing apps. A package may include custom tabs and objects and also custom links, custom profiles, reports, dashboards, documents, and more.

◇ **Managed Package:** These are used by Salesforce partners to distribute and sell applications to customers. These packages must be created from a Developer Edition organization.

◇ **Extension Package:** Package that adds functionality to a Managed Package.

◇ **Connected App:** This integrates an application with Salesforce using APIs. Connected apps use standard SAML and OAuth protocols to authenticate, provide single sign-on, and provide tokens for use with Salesforce APIs.

◇ **Namespace:** This differentiates your components from other components and allows you to do things such as upgrade the app after it's been installed. Within the underlying code, your namespace is prepended to all components that are packaged from your dev org.

◇ **Sharing:** Process by which you package a portion of your customizations and make them available, either publicly or privately, on the AppExchange.

◇ **Listing:** Primary way customers discover apps, components, and services to enhance their Salesforce experience.

◇ **Components:** It is one constituent part of a package. It defines an item, such as a custom object or a custom field. You can combine components in a package to produce powerful features or applications. In an unmanaged package, components are not upgradeable. In a managed package, some components can be upgraded while others can't.

◇ **Attributes:** It is a field on a component, such as the name of an email template or the Allow Reports checkbox on a custom object. On a non-upgradeable component in either an unmanaged or managed package, attributes are editable by both the developer (the one who created the package) and the subscriber (the one who installed the package). On an upgradeable component in a managed package, some attributes can be edited by the developer, some can be edited by the subscriber, and some are locked.

Building AppExchange Applications

Overview

Neither Free Edition nor Enterprise Edition is to be used if you are developing a custom application to sell to others. Developer Edition allows you to create Managed Packages which is the core Force.com vehicle for distributing applications. You can build a custom app with Developer Edition and, when you're ready for production, privately share it with your Enterprise or Professional Edition org.

You can create orgs from the Environment Hub for development, testing, and trial purposes. If you're an ISV partner, you can also create partner edition orgs with increased limits, more storage, and other customizations to support app development. You can create up to 20 member orgs per day. An org can be a member of only one Environment Hub at a time. From the Environment Hub, you can:

- Connect existing orgs to the hub with automatic discovery of related orgs.
- Create standard and partner edition orgs for development, testing, and trials.
- View and filter hub members according to criteria that you choose, like edition, creation date, instance, origin, and SSO status.
- Create single sign-on (SSO) user mappings for easy login access to hub members.

To upload an app or Lightning component to the AppExchange or to deploy changes between orgs, a **package** must be created. Once you release a package by publishing it on AppExchange, anyone can install it. When you create a package, the framework automatically detects dependent components and adds them to the package. Packages consist of one or more Salesforce components, which, in turn, consist of one or more attributes. Some related components (E.g. including custom object adds its custom fields, page layouts, and relationships with standard objects) are automatically included in the package even though they might not display in the Package Components list. If you create an app that uses components that aren't packageable, your subscribers will have to create and configure those components after they install your app.

Application that you plan to create could be

- -100% Native on force.com: Primary data, logic and user interface on Force.com. May callout to approved 3rd party web services (e.g. Google, Amazon, Facebook).
- Composite app: Primary data, logic run on something other than the Force.com. Application is an integration between salesforce.com and another applicationClient desktop or mobile app: No usage of partner hosted servers

Native Force.com and Composite Apps must create a package to begin the listing process and distribute it on the AppExchange.

Basic steps involved in package creation for these apps are:

- Create your custom app
- Create your Managed or Unmanaged Package
- Upload your Package to the AppExchange
- Register your Package to generate your listing
- You will receive an email that includes an installation link when your package has been uploaded successfully.

Client Apps never have a package as the app lives outside Salesforce and uses the API to integrate.

Unmanaged packages are traditionally used for distributing open-source projects to developers, or as a onetime drop of applications that require customization after installation.

When dealing with Custom App

Salesforce has three seasonal releases, all of which can impact any custom code you've developed internally. AppExchange vendors test their apps against new Salesforce releases well in advance of the actual release date so that they can resolve any conflicts close to Salesforce's release schedule. Often software vendors include upgrades in the monthly subscription price so re-installing from their AppExchange listing lets you use all the new features, at no additional cost. Typically, the cost of new feature development, support, support materials, and documentation are featured into the cost of their license fees.

When dealing with data changes – Internal and External

You might want to update your Salesforce data when changes occur in another service. Likewise, you might also want to update the data in another service (outside of Salesforce) based on changes to your Salesforce data. Salesforce provides ways of doing both transactions. An app built on the Force.com platform can connect with an external service in many ways.

- Create a custom link or custom formula field that passes information to an external service
- Use the Force.com API to transfer data in and out of Salesforce
- Use an Apex class that contains a Web service method

Beta version

Before you upload a production version of your app, it's a common practice to upload a beta version for testing. Your app isn't available to the public, but you can access it through an install link. The purpose of a beta is for testing only. Therefore, a beta can only be installed in a test org, Developer Edition, or sandbox. Installing the beta is easy, just click the link and provide the username and password you use for your test org. Beta packages can also be installed in sandboxes.

API and Dynamic Apex

By default, Dynamic Apex can only access the components with which the code is packaged. To provide access to standard objects not included in the package, the developer must set the API Access.

Connected Apps

Connected apps can be added to managed packages only. Subscribers or installers of a package can only uninstall (and not delete) its package. A developer can delete a connected app after a package is uploaded as Managed - Released. The connected app will be deleted in the subscriber's organization during a package upgrade. This is a packageable component. You can only package a connected app from a Developer Edition org.

The connected app has two modes of deployment.

- The app is created and used in the same org. This is a typical use case for IT departments.
- The app is created in one org and installed in other orgs. This is a typical use case for ISVs and entities with multiple org

Some Salesforce client app that are implemented as connected apps are Salesforce1 and Salesforce for Outlook. The Force.com IDE and Force.com Migration tool are “wrapper” connected apps that use the SOAP API to connect to Salesforce instead of OAuth, which other types of connected apps do.

To prevent naming conflicts, Salesforce recommends using managed packages for all packages that contain Apex. Managed packages receive a unique namespace. This way, all the Apex objects contain your namespace prefix.

To prevent naming conflicts, Salesforce recommends using managed packages for all packages that contain Apex. Managed packages receive a unique namespace. This way, all the Apex objects contain your namespace prefix.

You can also easily uninstall apps by choosing Setup ⇨ Exchange ⇨ GetApps and clicking the Uninstall link next to an app. Remember that if you uninstall, you delete, as a result, any custom object records related to the custom app.

Release States

Releasing an AppExchange package can be iterative, hence ensure each component functions as planned. You may even have beta testers who install an early version of your package and provide feedback.

A package helps you distribute your offering to other Salesforce users and organizations, including those outside your company. The package undergoes several states during the release process. Next, we discuss these states that Salesforce automatically applies to your package and components depending on the upload settings you choose and where it is in the release process.

Unmanaged Packages: The package has not been converted into a managed package or the component has not been added to a managed package. This is completely Editable by the Developer and Installer, but can NOT be upgraded. These packages are used for 1:1 Distribution, as Templates or one-offs and are also great for code sharing.

Managed Packages: These are typically used by Salesforce partners to distribute and sell applications to customers. These packages must be created from a Developer Edition organization. These packages ensure Intellectual property protection for Apex and seamless upgrading using patches. These packages are used for 1: Many Distribution, and typically used with Commercial Intent.

Managed Extensions: This package is used to add functionality to a Managed Package. It requires the Base MP is installed. It can be used for releasing Add-Ons and for supporting multiple editions.

Managed – Beta: The package or component was created in the current Salesforce organization and is managed, but it is not released either because it has not been uploaded or it has been uploaded with Managed – Beta option selected. This option prevents it from being published, publicly available on AppExchange. The developer can still edit any component but the installer may not be able to, depending on which components were packaged. This can only be installed in DE or Sandbox and can be modified. Also, this package can neither be upgraded, nor can have an AppExchange listing

Note:

Components and their attributes behave differently in managed and unmanaged packages.

Don't install a Managed - Beta package over a Managed - Released package. If you do, the package is no longer upgradeable and your only option is to uninstall and reinstall it.

A component that is "Managed - Beta" can become "Unmanaged" if it is removed from a managed package. All packages are unmanaged unless otherwise indicated by one of the managed icons.

Managed – Released: The package or component was created in the current Salesforce organization and is managed. It is also uploaded with the Managed - Released option selected, indicating that it can be published on AppExchange and is publicly available. Once you have moved a package to this state, some properties of the components are no longer editable for both the developer and installer. This can be installed in any org supporting the features. The package can be upgraded and certain components can be modified by developer and installer.

Patch: If you need to provide a minor upgrade to a managed package, consider creating a patch instead of a new major release. A patch enables a developer to change the functionality of existing components in a managed package, while ensuring that subscribers experience no visible changes to the package.

Managed – Installed: The package or component was installed from another Salesforce organization but is managed.

Publishing process

Once you join the Partner Community, you gain access to the Publisher Console/ Publishing page. This is where you declare, customize and track your contributions to the AppExchange.

All your app and service listings must be published under a single salesforce.com organization, known as your master AppExchange Publishing Organization (APO). The Developer edition organization where you create apps should not be designated as your APO.

High level steps involved in the publishing process are:

- a. Login to your dev org. Once your developer org is tied to the Publisher Console, select your package and sign a publisher agreement. This is where you will decide whether your component is free or not.
- b. Connect the organization where you developed the app or component to the AppExchange publishing console. To connect an organization, open the Publishing page in the Partner Community, and click the Organizations tab.
- c. Create your Provider Profile from the Company Info tab in the Publishing page in the Partner Community. The provider profile that you create includes the information about your company that you want to make visible to customers.
- d. As discussed above, your updated code is to be deployed to package environment
- e. You are directed to the Listings tab on the Publishing page. Create listing from Listings tab on the Publishing page in the Partner Community, so that other people can find your app and know what it does.
- f. Make the listing effective. As you craft your listing, you might want to use the guidelines as stated [here](#). You can add screenshots and videos (in the Media tab) to draw attention to key features, or add white papers to help demonstrate business value.

- g.** Finally, you want to connect your app to your business org so you manage the licenses for people that install your app.
- h.** You can also provide a free trial to potential customers to let them experience your offering before purchasing or subscribing.
- You need to create a Trialforce template and submit it for security review. This review is free and takes less time than the initial review of your app or component.
 - To link your Trialforce template to your AppExchange listing, go to the Listings tab. Find the listing where you want to offer a trial.
 - Click it to open the AppExchange publishing console. From the Trials tab you can offer a free trial.
 - A trial template gets added to the listing.
- i.** Register your package from the Packages tab and setup LMA. The License Management App (LMA) helps you manage sales, licensing, and support for your offering. The LMA comes preinstalled in your business organization. In this step, you connect your app to the LMA. This feature is available to eligible partners.
- j.** Right now, your app has a private listing on the AppExchange that you can share with potential customers, but the public doesn't see it unless they have the link. You can let customers try your offering in their own organizations or set up a test environment that you've customized. Before you can list the app publicly, you'll need to pass a security review.
- k.** Submit for security review by clicking 'Start Review' on the Packages tab.
- l.** Publish on AppExchange

Security Review Preparation

- a.** Guidelines, Checklists and other Resources to refer to
- Read the Force.com [Requirements Checklist](#) carefully.
 - [Secure Coding Guidelines](#) is a good document on various security risks from force.com standpoint.
 - Review the free resources listed on our [Secure Cloud Development](#) site.
 - Review the [OWASP Top Ten Checklist](#).
- b.** The [Trust Academy](#) offers practical, hands-on experience in the fundamentals of application security on the Force.com platform, how-to guidance on penetration testing tools like ZAP and Burp, and detailed instructions for how to submit for Security Review. You might also want to take the Security Review Trail head Module.
- c.** Scanning:
- Self-Service Source Code Analysis: [Source Scanner Portal](#) can be used by AppExchange Partners developing integrations for the AppExchange. This can be used to perform a static analysis scan of all unpackaged code in your organization against code developed on the Force.com Platform. When you [enter](#) the username of your Development org, all the force.com component (Apex, visualforce Page) will be scanned. Analyze this report, and act on all the problem points highlighted by the report. This free service is a partnership with Checkmarx. Once the issues are addressed and you get a clean chit by Checkmarx, proceed to the next tool. If Checkmarx returns anything besides "code quality" issues, you must either resolve or provide a document detailing false positives.
 - Scan External Integrations and Web Application Security
 - i) [Web Application Security Scanner](#) is a multipurpose web application security tool is based on a paid application called Burp Suite. It is recommended to create a new developer org and install the managed packed in the new developer Org. Once Burp scanner completes the scanning process, the report will be generated for the App in .html format. False positive report is created for all the medium and low issues. To know more about the Burp Security process, check [here](#).
 - ii) Run a free Web Application Security Scan with [Chimera](#) to help secure future AppExchange-listed web applications. Once your web-based application is ready to be scanned, use Chimera to take care of figuring out how to log in to your application and run a battery of different tests and scans. After the scan, a consolidated report with all issues, warnings, and informational notes is emailed to you.
 - iii) You can also set up ZAP locally to run a Web Application Security Scan against any external web application that is integrated with Force.com.

d. To seek help from Salesforce about the Submission Process and Security Review, sign up [here](#).

e. Testing

If you create sample records for your custom objects, you figure it out quickly, if you're missing something. Then a good trick is to clone records by clicking the Clone button on the record's detail page. For importing (adding and loading) data, AppExchange Data Loader and the Excel Connector are available for free to Enterprise Edition customers. Professional Edition customers now can use Excel Connector.

- You can test the beta package in a partner testing organization (Enterprise, Professional or Group Editions are available).
- a. Manual and Automated code review, scan and testing: Manually test your app to ensure it meets review requirements not found by tools. You can use the [OWASP Testing Guide for help](#).
- Architecture review
- Web server testing

Note: *If you are not sure if something is in scope (e.g. credentials for web services), include it anyway. Provide access to any environments, packages, and external components your app uses and include any documentation that comes with your product. If the Security Review team determines something is missing there will be delays. After all the above have been taken care of, you can submit your app for Security Review. You will receive an email within 48 business hours confirming submission. Please retain this email as evidence of your submission with the time/date stamp.*

Security Review Preparation

The review process takes about 6-8 weeks from when you have everything provided

- Your documentation is complete and accurate
- The test environment is complete, fully configured, and includes all necessary information
- You have met the requirements
- You are within the agreement guidelines

Before submitting for security review, make sure your application is enrolled in either the [ISVforce or Force.com Embedded program](#).

Initiate security review for your offering by logging into the [AppExchange Publishing Console](#) and clicking "Start Review" on your offering. For existing offerings that are due for a security re-review, you must also submit a case in the [Partner Portal](#).

The steps involved in the process are:

- Provide Company Information
- Define Application Components: Such as force.com components, technologies used, any third party APIs, web service or web application, and others. Provide Test Environments
- Provide Test Environments
- Upload Scanner Reports: You would be prompted to submit the security reports generated earlier. In "Force.com Security Code Scanner report", attach the report you got from Checkmarx. If Checkmarx reported any false positives, attach a document explaining each one. If you are using external services, you would have used ZAP or Chimera to scan them. Provide any reports from these tools in "Other report or documentation".
- Payment: You may also have to pay for the security review and the annual listing fee (discussed in the section 'Pricing' above).
- Distribution agreement with Salesforce

Once you have submitted your app for Security review, you need to submit a case on Salesforce Support. This will expedite your review process. Manual and automated application and network security testing will be performed by the security review team and you will receive your review results.

Salesforce's security review process is both qualitative (question and answer round-to-review policies and procedures) and quantitative (network and application penetration tests).

Partner applications are subject to periodic, random re-review at any time.

Acting on Review results

The security review process improves the quality of your product and increases your chances of a successful launch. The main reasons why an application might fail a security review can be found [here](#).

- a. Approved: Now you can list your offering publicly on the AppExchange and distribute it to customers.
- b. Provisionally approved: Medium and/or low-risk issues have been identified. You will be given a mutually acceptable timeline to address these issues (usually 60-90 days). In the meanwhile, you will be allowed to list your application on AppExchange Security Review Results. If the issues are not addressed within the given timeline, the app may be removed from AppExchange.
- c. Not approved: Critical and/or high-risk issues have been identified. You get this feedback as a report that lists the vulnerabilities that the security team found. The email you receive also has detailed instructions on how to fix these vulnerabilities.

As you fix the vulnerabilities, don't forget to reuse scanners and adversarial testing on your product, just as you did before the review. They help prevent new vulnerabilities from sneaking into your code.

Publish

Once you have passed the security review you may login to the AppExchange and make your listing live. You can also request an API token for your app after it passes the AppExchange security review. With the token, it can use the SOAP and REST APIs.

Using the AppExchange and the License Management Application (LMA), you can sell, renew and manage user-based licenses to the app and set default license settings. For example, you can license your app as a free trial that expires after a specified number of days.

Salesforce.com reserves the right to conduct random on-site and off-site tests on published offerings. If during these tests, Salesforce finds that the offering has deviated from any of the requirements, you will be notified and provided a timeframe to remedy the issue. In extreme cases, the AppExchange listing may be pulled from public viewing.

Press and Publicity

While you are waiting for the security review results, you can prepare your project's listing. This is where you will describe your app's features, supported devices, licensing options and provide interesting descriptions to boost installs.

Once your listing is published, use your preferred social media to advertise your work and monitor some analytics such as the number of installs in the Publisher Console. You will also want to monitor user feedback as they post reviews, request additional information or declare bugs.

Other Resources

[Overview of Publishing App on AppExchange](#)

[Security Development Center](#)

[Requirements Checklist Security Review](#)

[Security Review Trailhead Module](#)

[Security Office Hours](#)

[Security Review FAQ](#)