

Windows App Workshop

Hands-on lab

Workbook 2: To the Windows Store

Lab version: 2.0.0
Last updated: 10/19/2013

CONTENTS

OVERVIEW3

EXERCISE 1: GET A WINDOWS DEVELOPER ACCOUNT5

 Task 1 – Create Your Account 5

EXERCISE 2: PREPARE YOUR APP5

 Task 1 – The App Manifest 5

 Task 2 –App Logos..... 8

EXERCISE 3: BEGINNING THE STORE SUBMISSION.....9

 Task 1 – Associating Your App with the Store..... 9

EXERCISE 4: CREATE AN APP PACKAGE 15

 Task 1 – Create the App Package 15

EXERCISE 5: COMPLETE YOUR APP SUBMISSION23

 Task 1 – Complete the App Submission 23

SUMMARY26

Overview

Welcome to the **Windows App Workshop**! This series will help you build useful Windows and Windows Phone apps based on data and services provided by third parties via their own APIs. These workshops feature starter kits in XAML/C# and HTML/JavaScript that will help you quickly get started using these APIs, submitting your apps to the marketplace and driving demand for your apps through a set of marketing activities.

These hands-on lab workbooks provide the prescriptive guidance on how to use these Starter Kits to create your very own Windows 8 and Windows Phone Apps.

- **Workbook 2 – Getting Your App into the Windows Store**

- Get a Windows Developer Account
- Using the Dashboard
- Submitting an App
- Addressing Certification Issues

As you work through these workbooks, you will learn to access data via APIs, create an interface with controls, bind that data to controls, and submit your app to the Windows Store. Along the way, you'll see Windows 8 and Windows Phone features to help your app shine, including sharing, searching, tiles, and more.

Objectives

This workbook will show you how to:

- Prepare your app for Windows Store submission
 - Create your app package
 - Upload your app package
 - Complete your app's submission
-

System requirements

You must have the following to complete this workbook:

- Microsoft Windows 8
 - Microsoft Visual Studio 2012 (any version)
-

Exercises

This Hands-on workbook includes the following exercises:

1. Get a Windows Developer Account
 2. Prepare Your App
 3. Beginning the Store Submission
 4. Create an App Package
 5. Complete Your Submission
-

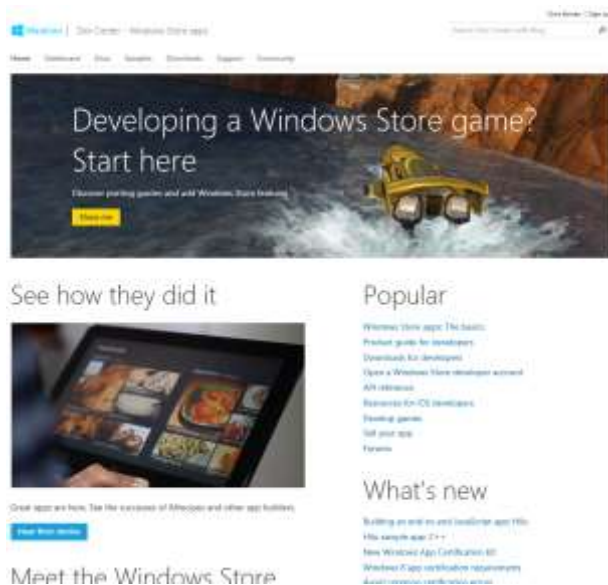
Estimated time to complete this workbook: **50 to 70 minutes.**

Exercise 1: Get a Windows Developer Account

If you haven't already, now is the time to register for a Windows Developer account. This will allow you to reserve app names and submit apps to the Windows Store.

Task 1 – Create Your Account

1. Go to <http://dev.windows.com>



2. Click *Dashboard* in the top navigation bar.
3. Sign in with a Microsoft ID and follow the instructions.

Note: Students (via DreamSpark at <https://www.dreamspark.com>) and paid MSDN subscribers have options to make this registration free.

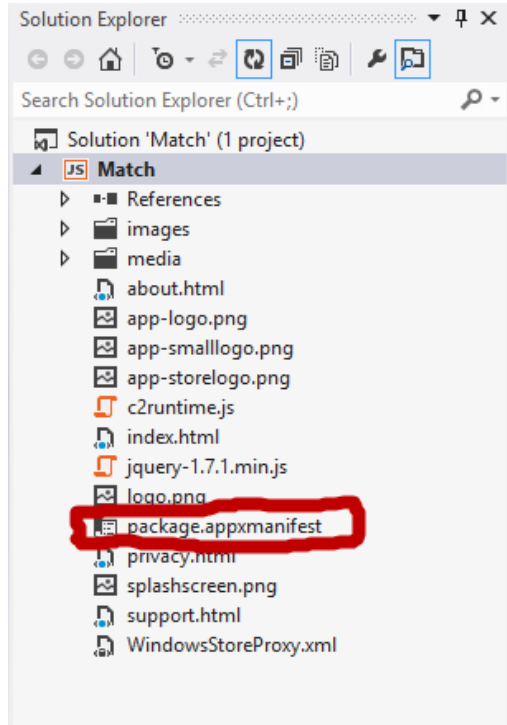
Exercise 2: Prepare Your App

In the exercise, you'll update your app to have the required settings for submission to the Windows Store.

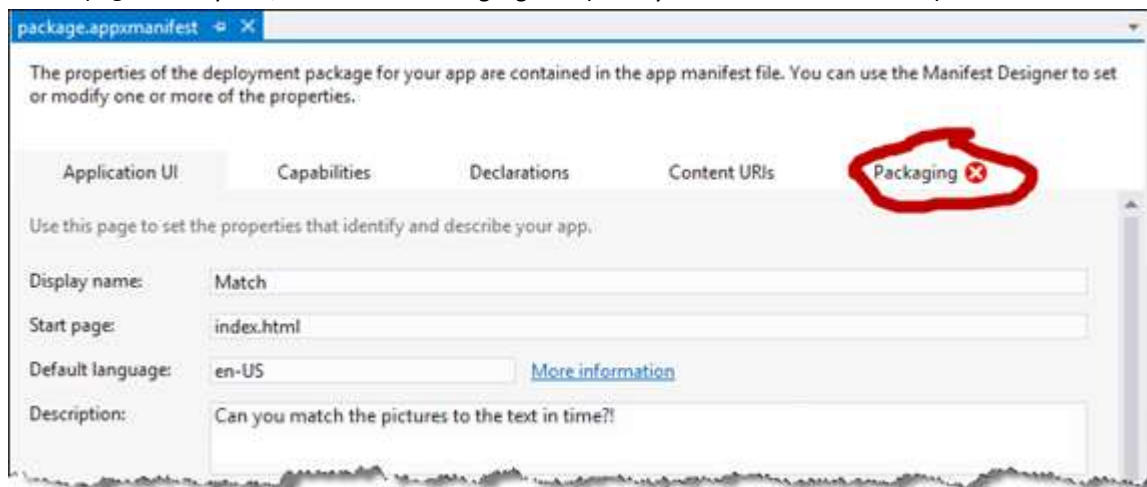
Task 1 – The App Manifest

Here, we'll open the app manifest and ensure the proper settings are in place before proceeding.

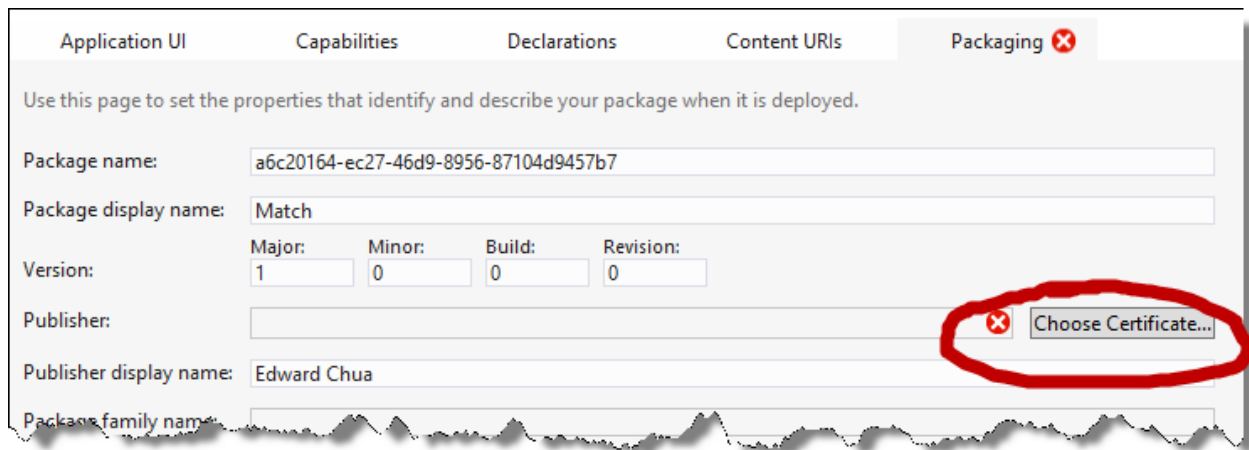
1. Open your project and double-click on the package.appxmanifest file:



2. In the page that opens, select the *Packaging* tab (it may have a red X next to it)...

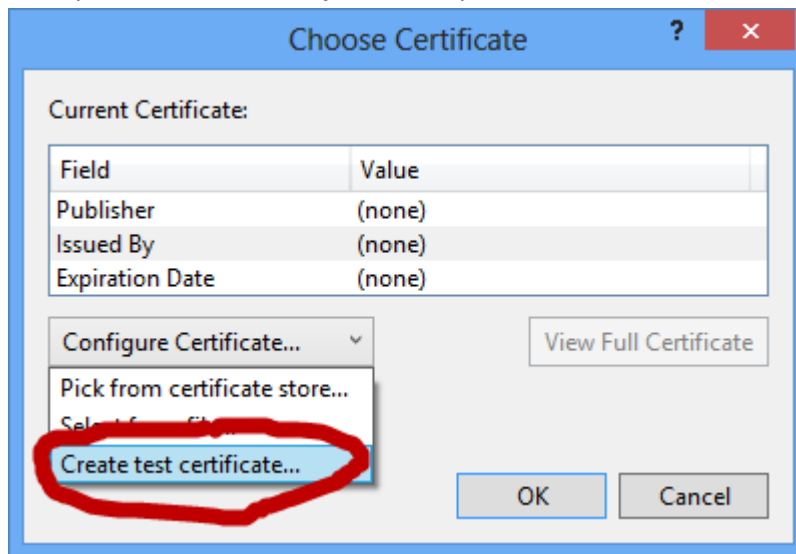


- On the Packaging tab, click the Choose Certificate button.



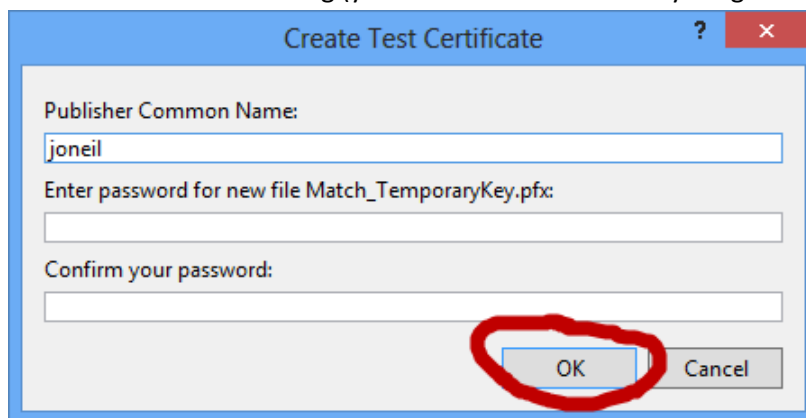
The screenshot shows the 'Packaging' tab of a software interface. The tab is highlighted with a red 'X' icon. Below the tab, there is a text box for 'Package name' containing 'a6c20164-ec27-46d9-8956-87104d9457b7'. Below that is a text box for 'Package display name' containing 'Match'. Below that is a 'Version' section with four input boxes: 'Major' (1), 'Minor' (0), 'Build' (0), and 'Revision' (0). Below that is a text box for 'Publisher' which is empty. To the right of the 'Publisher' text box is a button labeled 'Choose Certificate...' with a red 'X' icon next to it. Below that is a text box for 'Publisher display name' containing 'Edward Chua'. Below that is a text box for 'Package family name' which is empty.

- Then pick *Create Test Certificate*, and press OK.



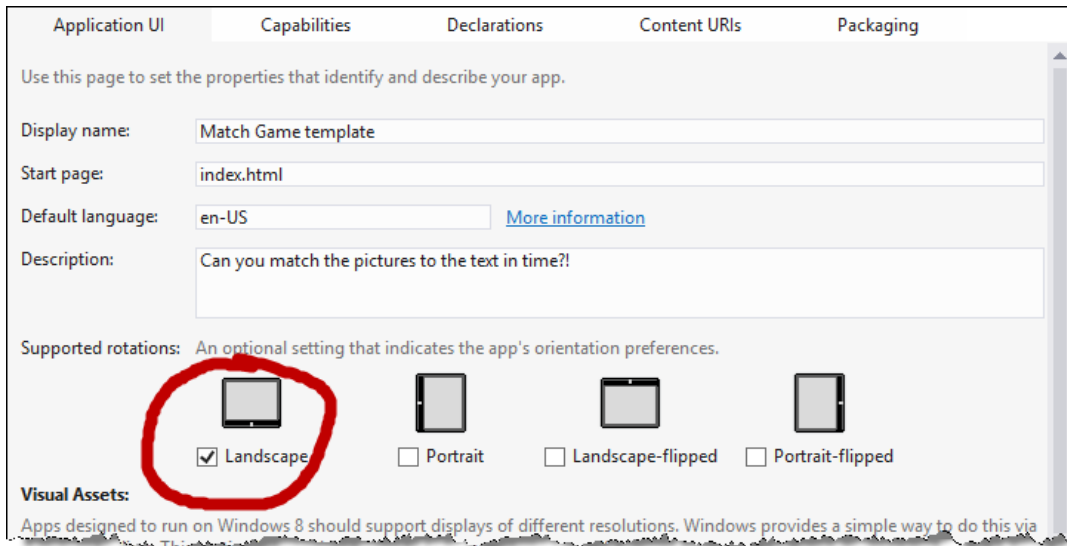
The screenshot shows the 'Choose Certificate' dialog box. It has a title bar with a question mark and a close button. Inside, there is a section 'Current Certificate:' with a table showing fields and values: 'Publisher' (none), 'Issued By' (none), and 'Expiration Date' (none). Below the table is a dropdown menu labeled 'Configure Certificate...' with a list of options: 'Pick from certificate store...', 'Select from file...', and 'Create test certificate...'. The 'Create test certificate...' option is highlighted with a red circle. To the right of the dropdown is a button labeled 'View Full Certificate'. At the bottom are 'OK' and 'Cancel' buttons.

- Press OK on the next dialog (you don't need to enter anything in the text boxes)

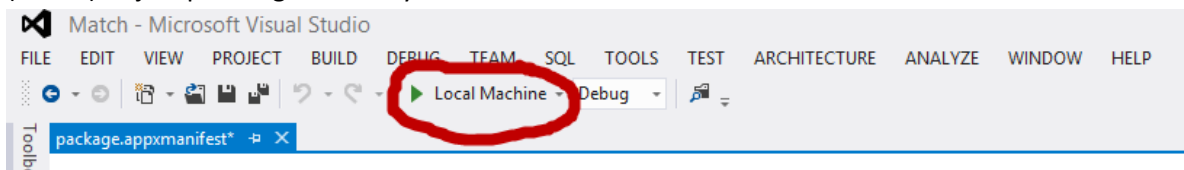


The screenshot shows the 'Create Test Certificate' dialog box. It has a title bar with a question mark and a close button. Inside, there is a text box for 'Publisher Common Name:' containing 'joneil'. Below that is a text box for 'Enter password for new file Match_TemporaryKey.pfx:' which is empty. Below that is a text box for 'Confirm your password:' which is empty. At the bottom are 'OK' and 'Cancel' buttons. The 'OK' button is highlighted with a red circle.

- Depending on the nature of the app you may decide to only support landscape orientation. If so you can make do so on the Application UI tab of the application manifest.



- You can now run your app as a full-fledged Windows 8 application by clicking the arrow next to Local Machine (below) or just pressing the F5 key.



- At this point run your app just to make sure it works as you expected. You should check out snapped view (move mouse to top of screen until you see the hand cursor, then drag to middle of screen and then all the way left). Your app will now occupy about a third of the screen.

Task 2 –App Logos

Before submitting to the store, you need to provide a minimum of four graphics – three logos and a splash screen. These graphics have very specific size requirements (see below) and should be .png or .jpg format. Use MS Paint or other free drawing tools to create these graphics.

- Create graphics for your app in the following sizes:

Asset	Filename	Size in Pixels
Logo	app-logo.png	150 x 150
Small Logo	app-smallogo.png	30 x 30
Store Logo	app-storelogo.png	50 x 50
Splash Screen	splashscreen.png	620 x 300

- Overwrite the existing file in the project directory (filenames above) with your custom graphics.

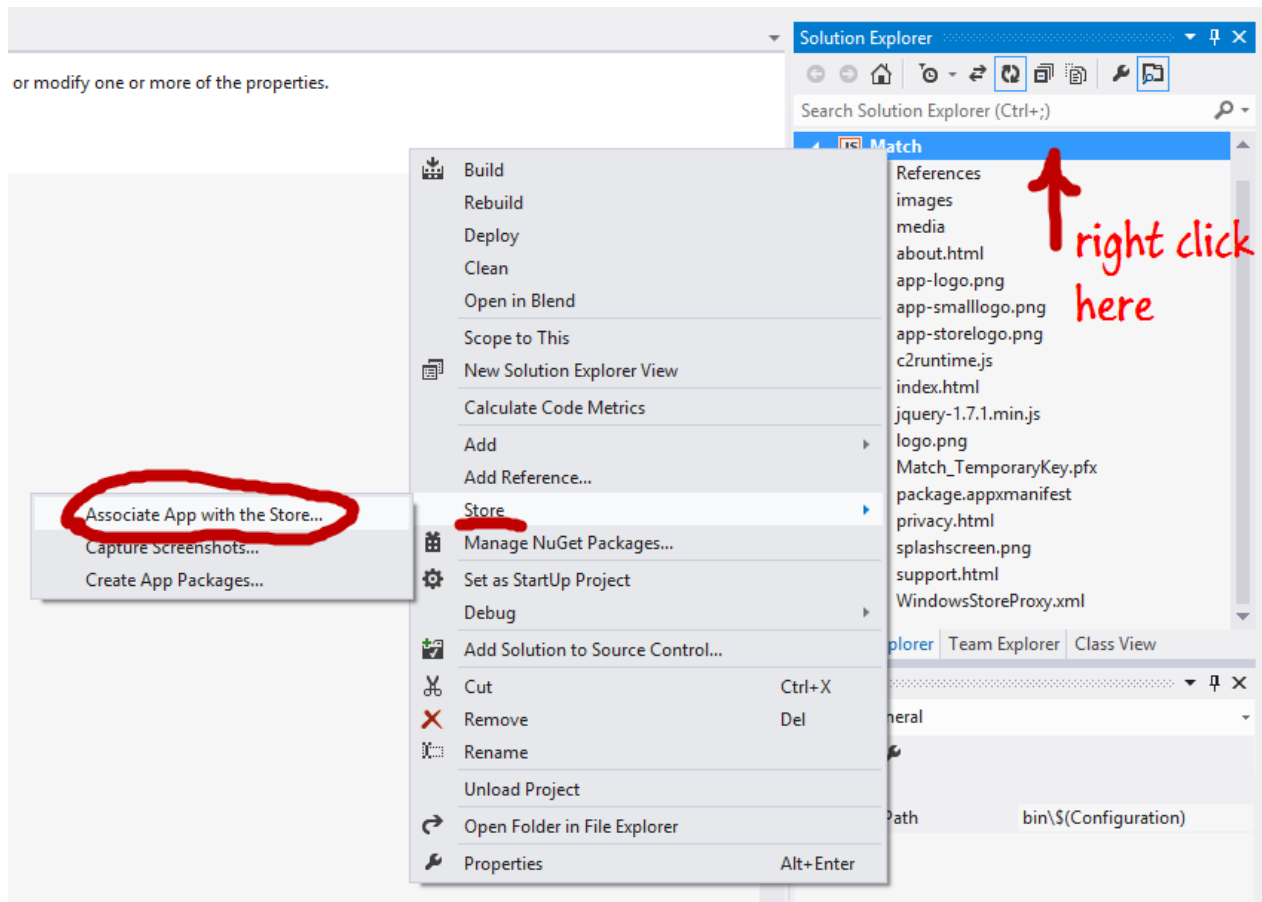
Exercise 3: Beginning the Store Submission

Now you're ready to begin the app submission process. To do this, you'll need the Windows Developer account described in Exercise 1.

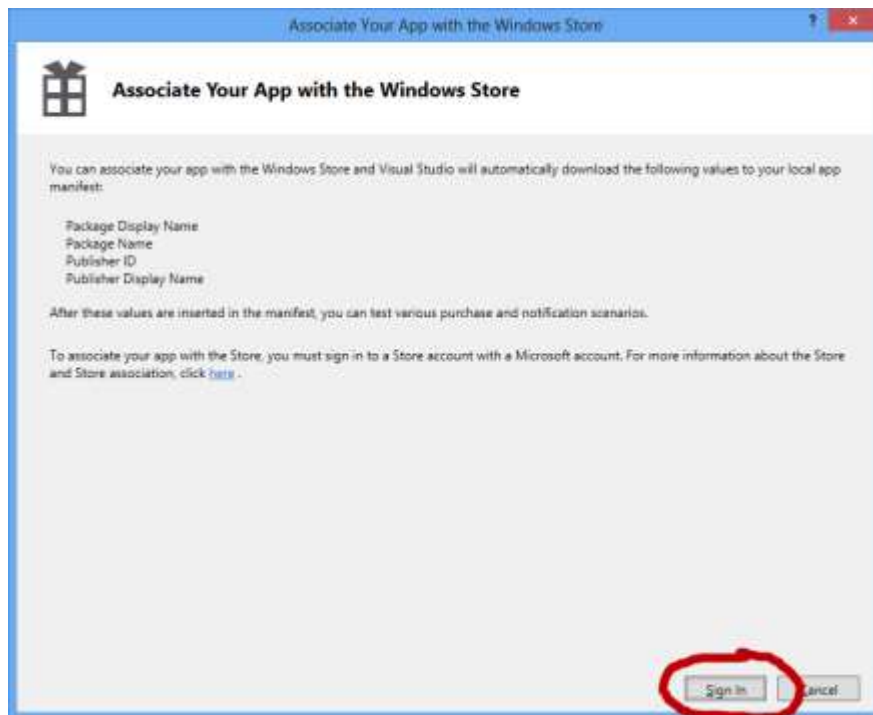
Task 1 – Associating Your App with the Store

Here, you will connect your app with a Windows Store app entry on your developer dashboard.

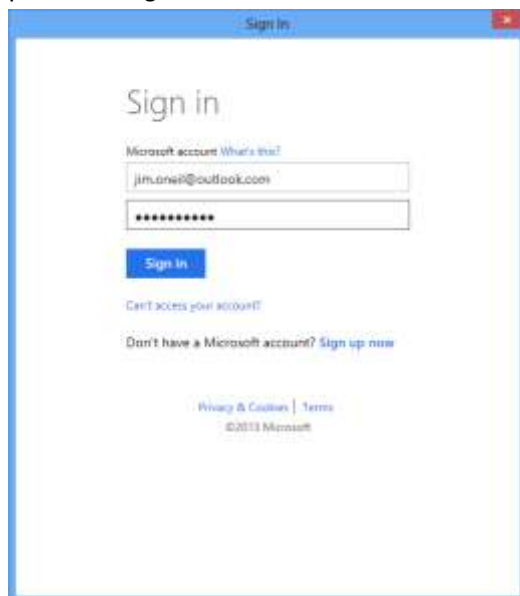
1. Right-click on the project in Solution Explorer to bring up the menu, then select *Store*, and *Associate App with Store...*



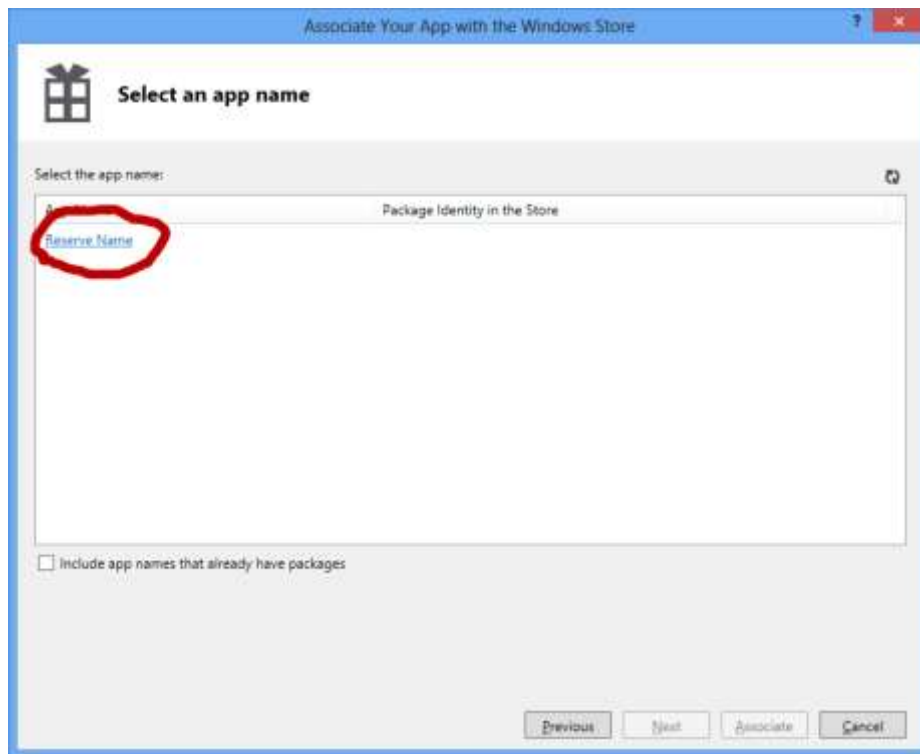
2. On the dialog that results, press the *Sign in* button



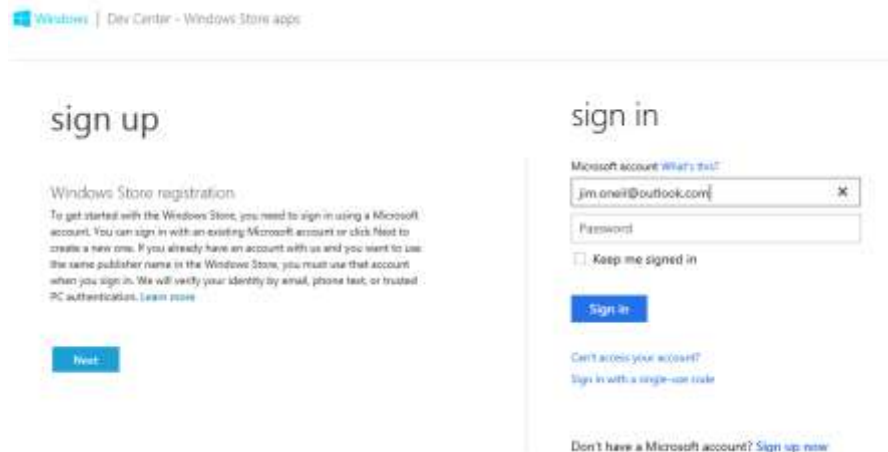
3. You will be prompted to enter your Microsoft account ID for your store account; enter your credentials and press the *Sign In* button



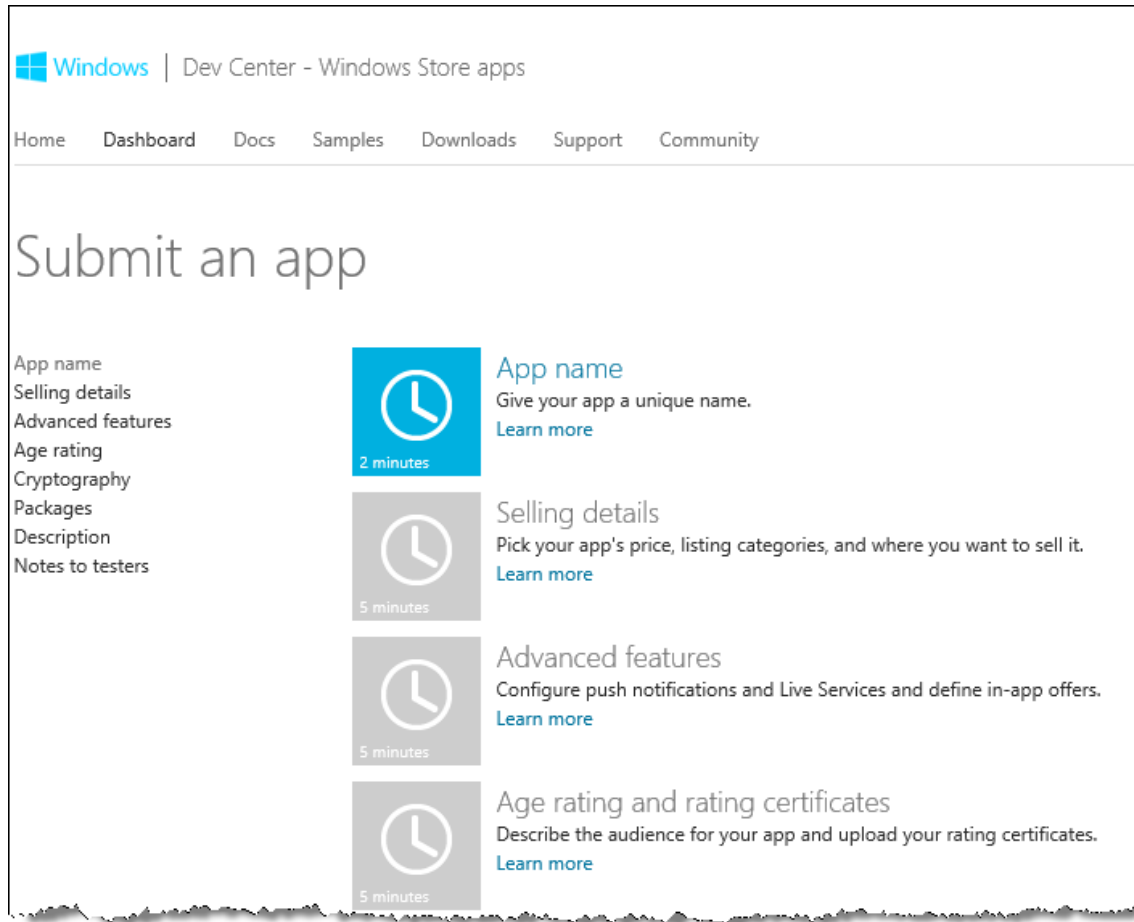
4. After you sign in, you'll see the following dialog. Click the *Reserve Name* link.



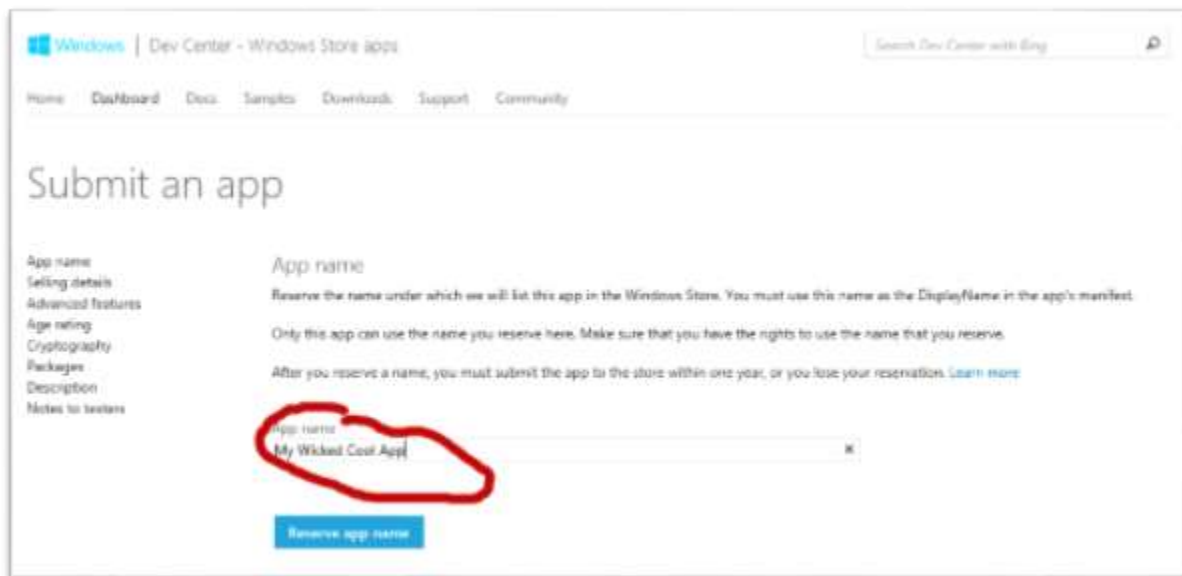
5. You'll be asked to sign in again



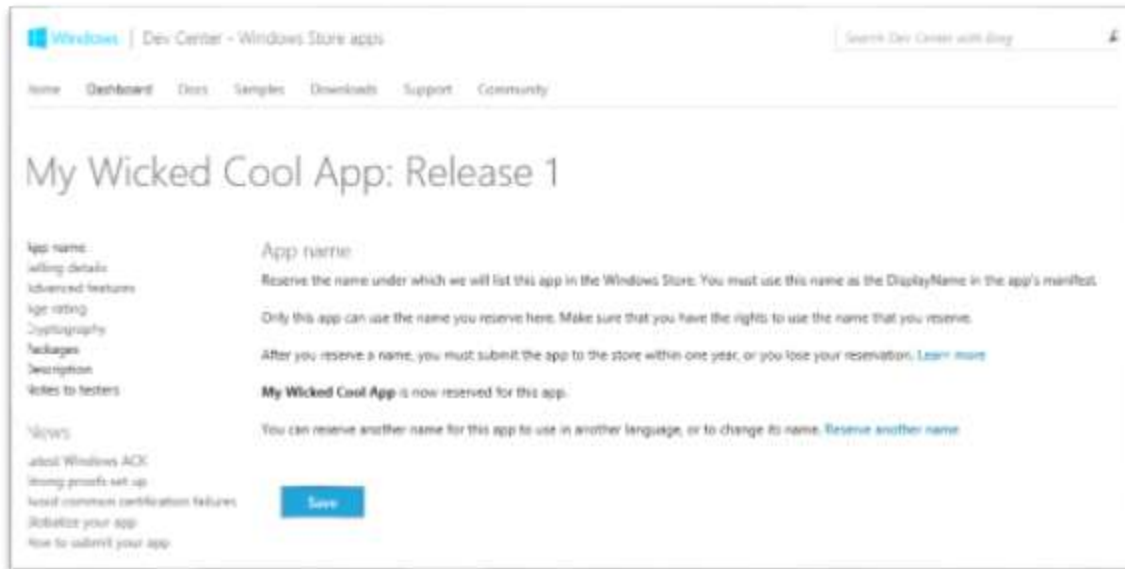
6. You'll then see the steps to submit the app to the store.



7. App name: pick a name for your app. It has to be unique among all apps in the Store. When you click *Reserve app name*, the name will be checked for uniqueness.



8. Press the **Save** button



9. Continue by clicking **Selling details**

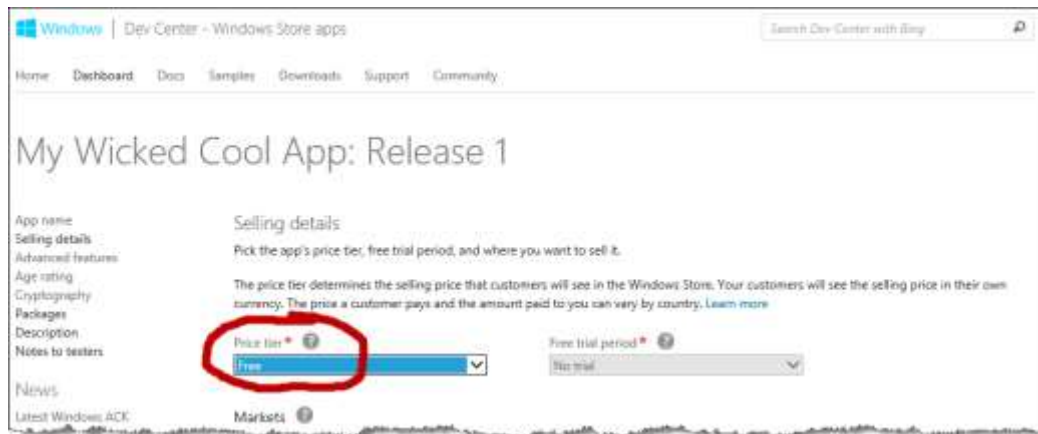


Selling details

Pick your app's price, listing categories, and where you want to sell it.

[Learn more](#)

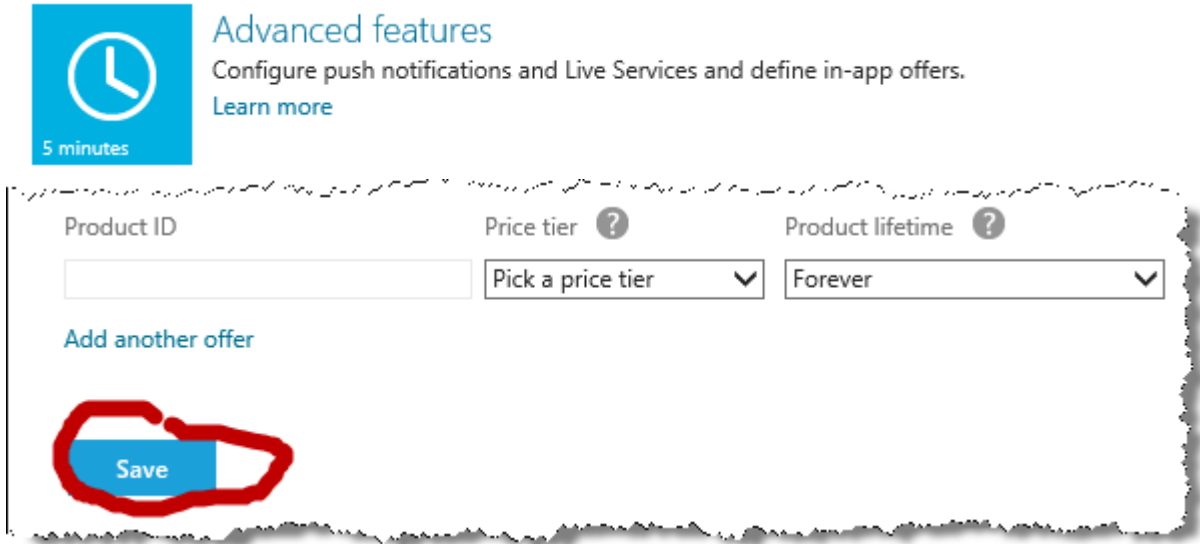
10. Select your price tier and trial period as you wish.



11. For the rest of the form, consider the following, then complete the form and press **Save**.

Markets:	For simplicity, you may want to start with only the United States. Other countries may have additional requirements, but see the documentation for details.
Release date:	Leave as <i>Release my app as soon as it passes certification</i> .
Category	Select a Category
Subcategory	Select a Subcategory (not all Categories will have subcategories)
Hardware Requirements	Leave both Minimum DirectX feature level and Minimum system RAM set to <i>Available to all systems</i> .
Accessibility	Leave this unchecked

12. Continue with **Advanced features**, do nothing except press the *Save* button



Advanced features
Configure push notifications and Live Services and define in-app offers.
[Learn more](#)

5 minutes

Product ID

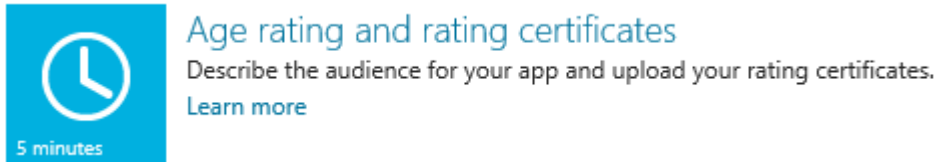
Price tier

Product lifetime

[Add another offer](#)

Save

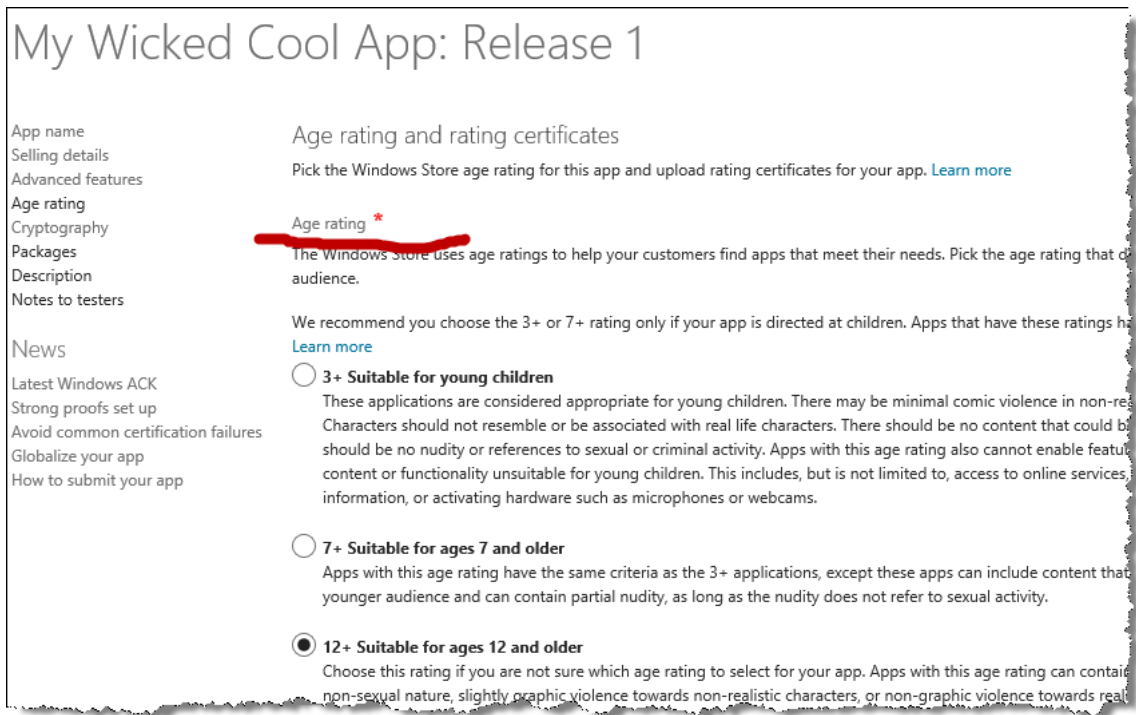
13. Continue with **Age rating and rating certificates**



Age rating and rating certificates
Describe the audience for your app and upload your rating certificates.
[Learn more](#)

5 minutes

14. Select an age rating for your application based on the app's content and capabilities, and press the *Save* button.



My Wicked Cool App: Release 1

App name
Selling details
Advanced features
Age rating
Cryptography
Packages
Description
Notes to testers

News
Latest Windows ACK
Strong proofs set up
Avoid common certification failures
Globalize your app
How to submit your app

Age rating and rating certificates
Pick the Windows Store age rating for this app and upload rating certificates for your app. [Learn more](#)

Age rating *
The Windows Store uses age ratings to help your customers find apps that meet their needs. Pick the age rating that is best for your app's audience.

We recommend you choose the 3+ or 7+ rating only if your app is directed at children. Apps that have these ratings have more restrictions. [Learn more](#)

☐ **3+ Suitable for young children**
These applications are considered appropriate for young children. There may be minimal comic violence in non-realistic situations. Characters should not resemble or be associated with real life characters. There should be no content that could be considered sexual or criminal activity. Apps with this age rating also cannot enable features that require internet access or functionality unsuitable for young children. This includes, but is not limited to, access to online services, location information, or activating hardware such as microphones or webcams.

☐ **7+ Suitable for ages 7 and older**
Apps with this age rating have the same criteria as the 3+ applications, except these apps can include content that is suitable for a younger audience and can contain partial nudity, as long as the nudity does not refer to sexual activity.

☒ **12+ Suitable for ages 12 and older**
Choose this rating if you are not sure which age rating to select for your app. Apps with this age rating can contain content that is suitable for a non-sexual nature, slightly graphic violence towards non-realistic characters, or non-graphic violence towards real people.

15. Continue with **Cryptography**



Cryptography

Declare whether your app uses cryptography and enable package upload.

[Learn more](#)

16. Verify whether your app does or does not use cryptography and continue by clicking **Save**.

Exercise 4: Create an App Package

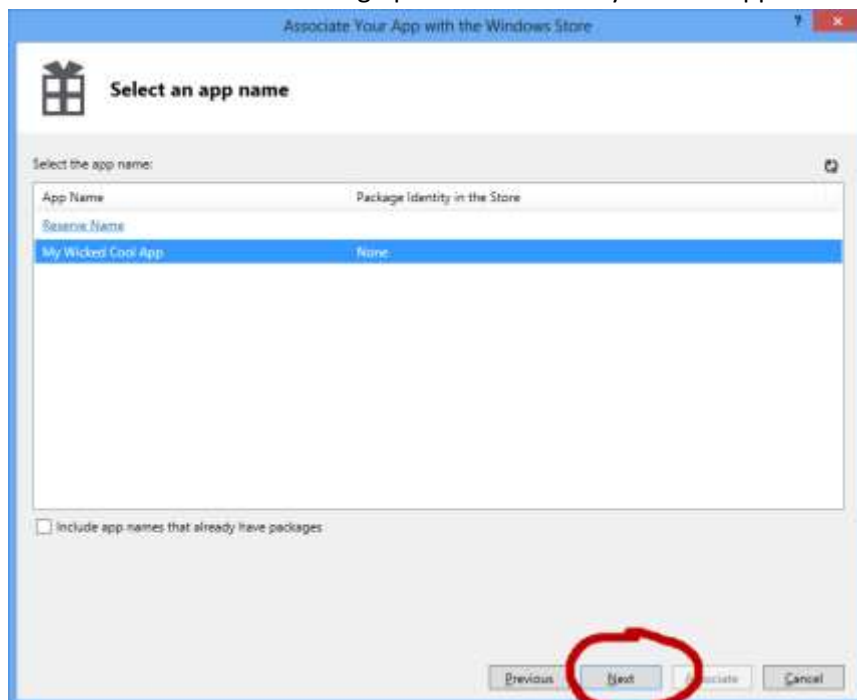
At this point you have created most of the application profile in your Store developer account! Next you'll create the app package for upload you'll go back to Visual Studio to create a package to upload your app to this new application profile.

Task 1 – Create the App Package

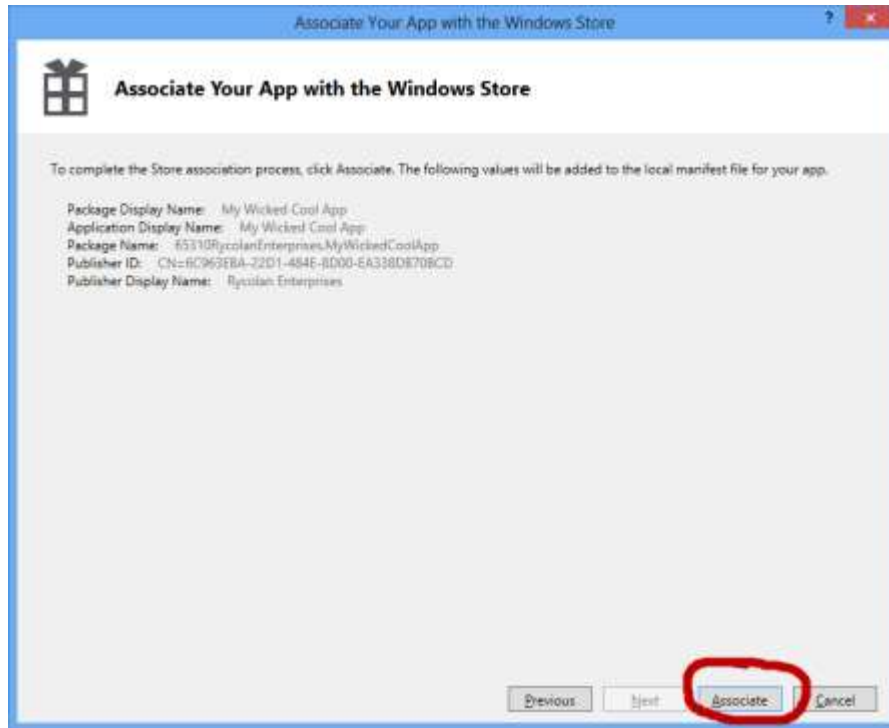
Next you'll go back to Visual Studio to create a package to upload your app to this new application profile.

17. Return to your Visual Studio project.

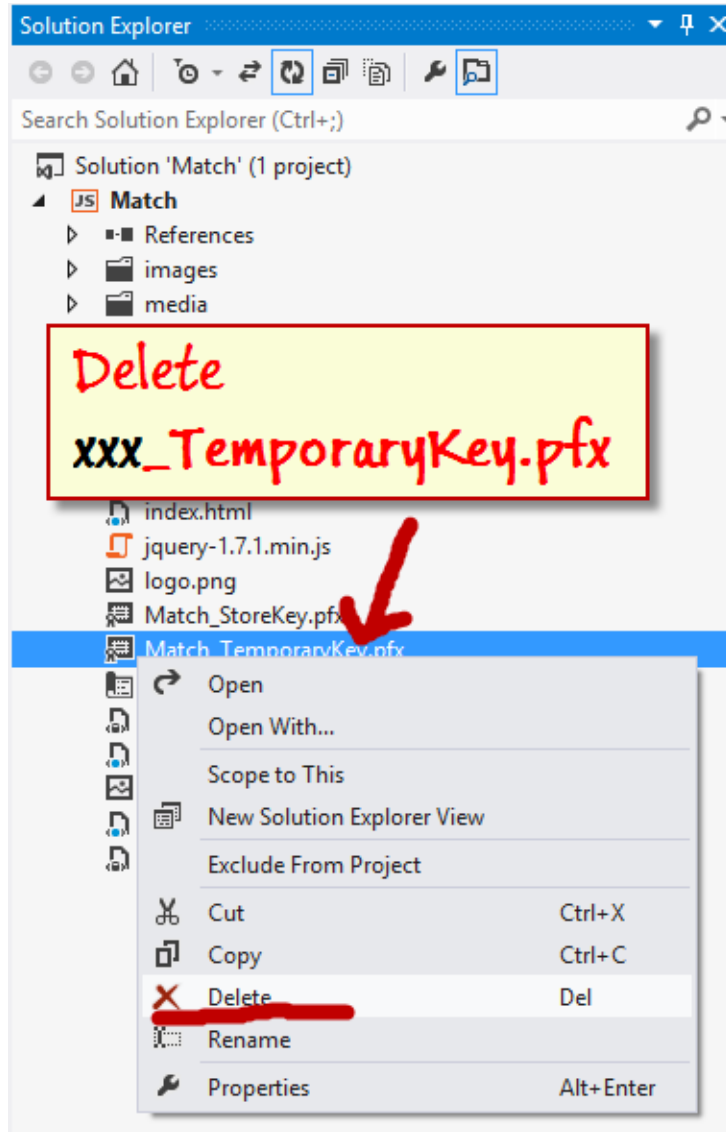
18. You should now see the dialog updated to include your new application. Press the *Next* button.



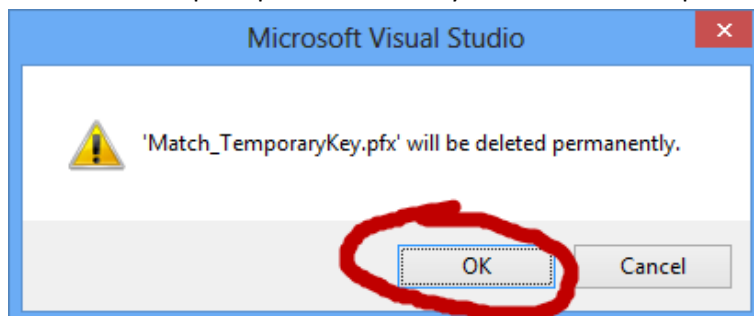
19. Now press the *Associate* button:



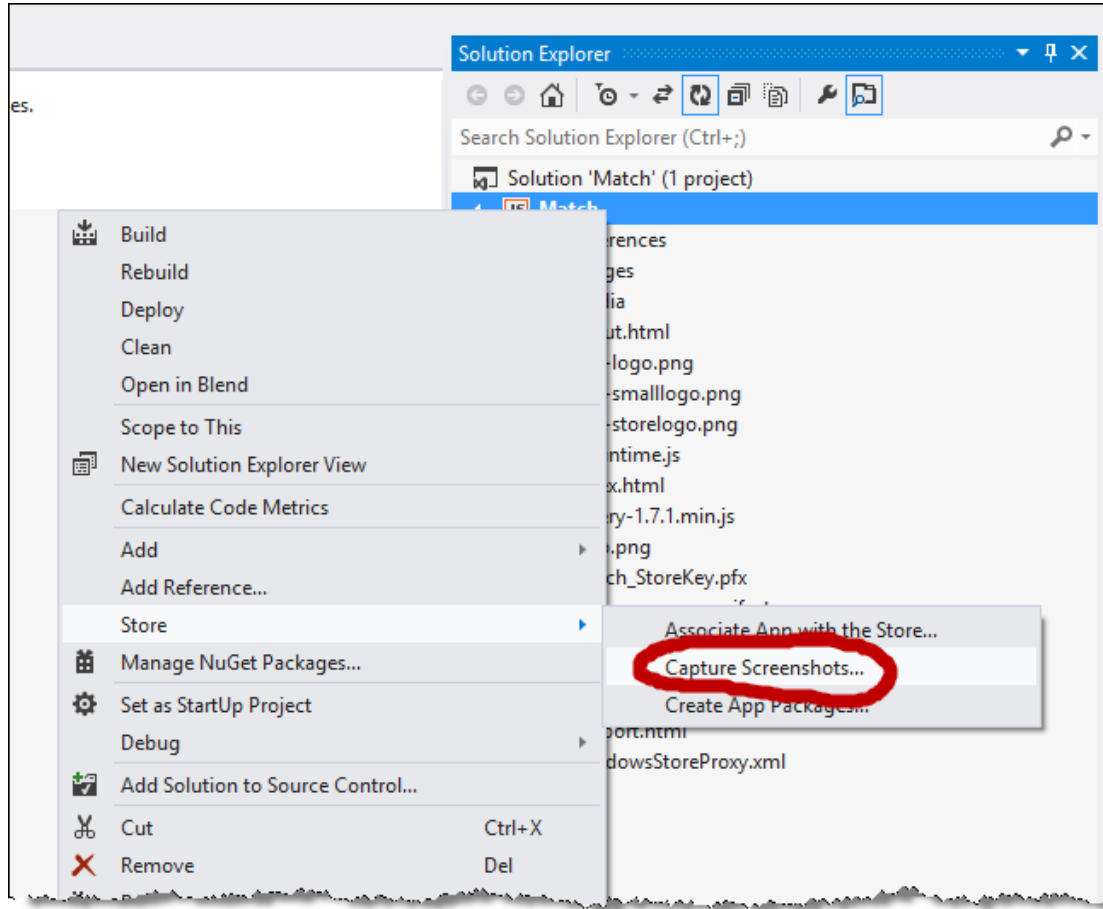
20. In Visual Studio Solution Explorer, delete the file ending in **TemporaryKey.pfx**, by selecting it and clicking the right mouse button to bring up the dialog, then clicking delete. **DO NOT DELETE** the one ending in StoreKey.pfx.



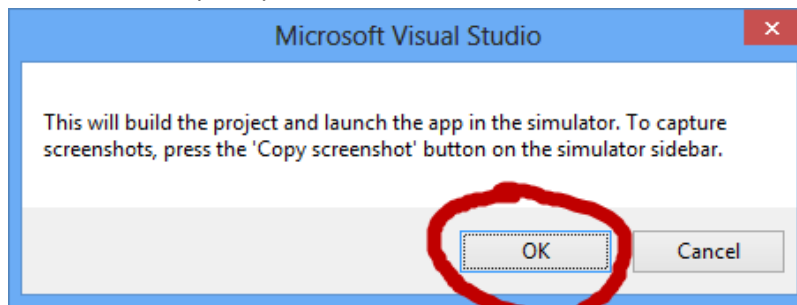
21. Press OK when prompted to confirm you want to delete permanently.



22. Select the *Store -> Capture Screenshots...* option by clicking the right mouse button on the project in Solution Explorer.



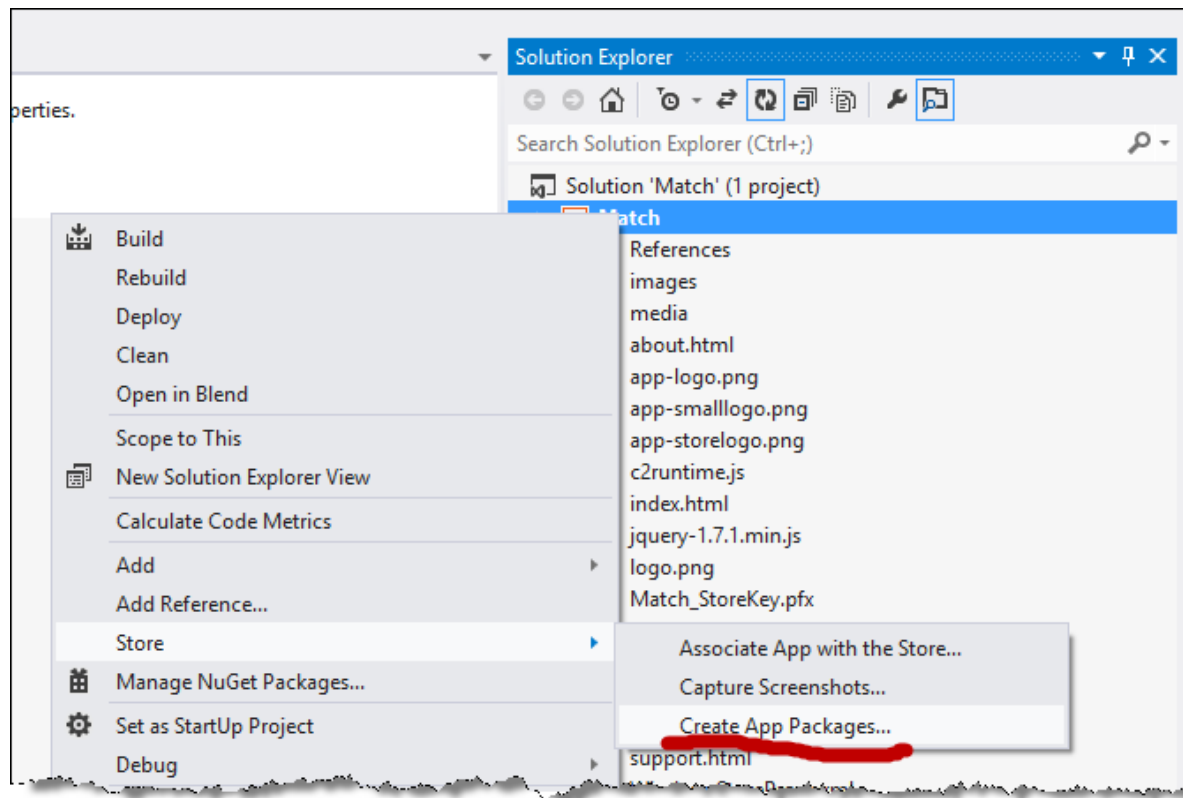
23. Press OK at the prompt:



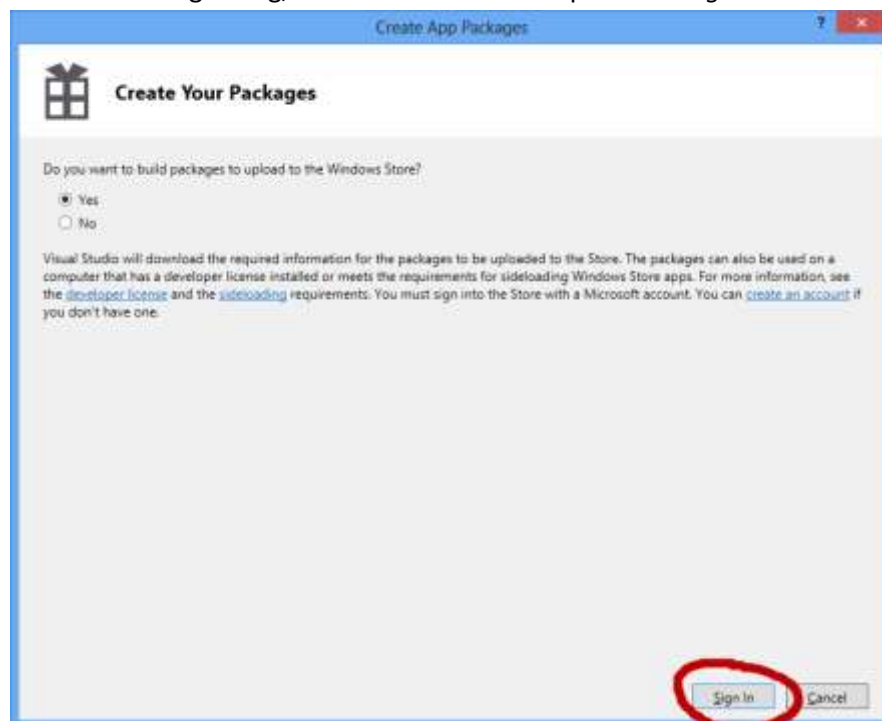
The Simulator should automatically start and begin running your app. Use the app to show features you want to highlight, then click the Camera button on the right-hand toolbar to take one or more screenshots.

Note: The screenshots will be stored by default in your My Pictures library, in a directory called Windows Simulator. You'll need these screenshots later as you complete your application profile in the Windows Store dashboard.

24. In Visual Studio, select the *Store -> Create App Packages* option



25. On the resulting dialog, leave *Yes* selected and press the *Sign in* button.



26. Sign into your developer account again.

Sign In

×

Sign in

Microsoft account [What's this?](#)

Sign in

[Can't access your account?](#)

Don't have a Microsoft account? [Sign up now](#)

[Privacy & Cookies](#) | [Terms](#)

©2013 Microsoft

27. Select your app from the list, and press the *Next* button.

The screenshot shows the 'Create App Packages' dialog box with the title 'Select an app name'. It features a gift icon and a 'Select the app name:' label. Below this is a table with two columns: 'App Name' and 'Package Identity in the Store'. The first row is highlighted in blue and contains the text 'My Wicked Cool App' and 'None'. There is a 'Reserve Name' link above the table. Below the table is a checkbox labeled 'Include app names that already have packages'. At the bottom right, there are four buttons: 'Previous', 'Next' (circled in red), 'Create', and 'Cancel'.

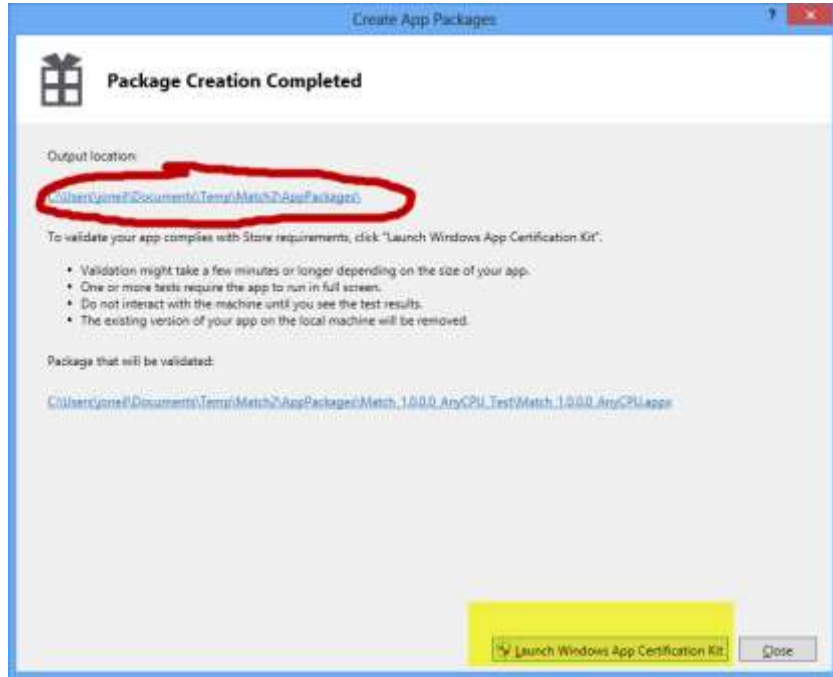
App Name	Package Identity in the Store
My Wicked Cool App	None

28. On the screen that follows, press Create

The screenshot shows the 'Create App Packages' dialog box with the title 'Select and Configure Packages'. It features a gift icon. The 'Output location:' field shows 'C:\Users\joneil\Documents\Temp\Match2\AppPackages\'. The 'Version:' field shows '1.0.0.0' with a checked 'Automatically increment' checkbox. Below this is a table for selecting packages to create and solution configuration mappings. The table has two columns: 'Architecture' and 'Solution Configuration'. The first row is checked and shows 'Neutral' and 'Release (Any CPU)'. The other rows are unchecked and show 'x86', 'x64', and 'ARM' architectures with their respective 'Release' configurations. At the bottom, there is a checked checkbox for 'Include public symbol files, if any, to enable crash analysis for the app'. At the bottom right, there are four buttons: 'Previous', 'Next', 'Create' (circled in red), and 'Cancel'.

Architecture	Solution Configuration
<input checked="" type="checkbox"/> Neutral	Release (Any CPU)
<input type="checkbox"/> x86	Release (x86)
<input type="checkbox"/> x64	Release (x64)
<input type="checkbox"/> ARM	Release (ARM)

29. On the next screen, click the link labeled *Output location*; that will open a File Explorer window that you should leave open.



30. Run the Windows Application Certification Test Kit to make sure your app meets tested requirements

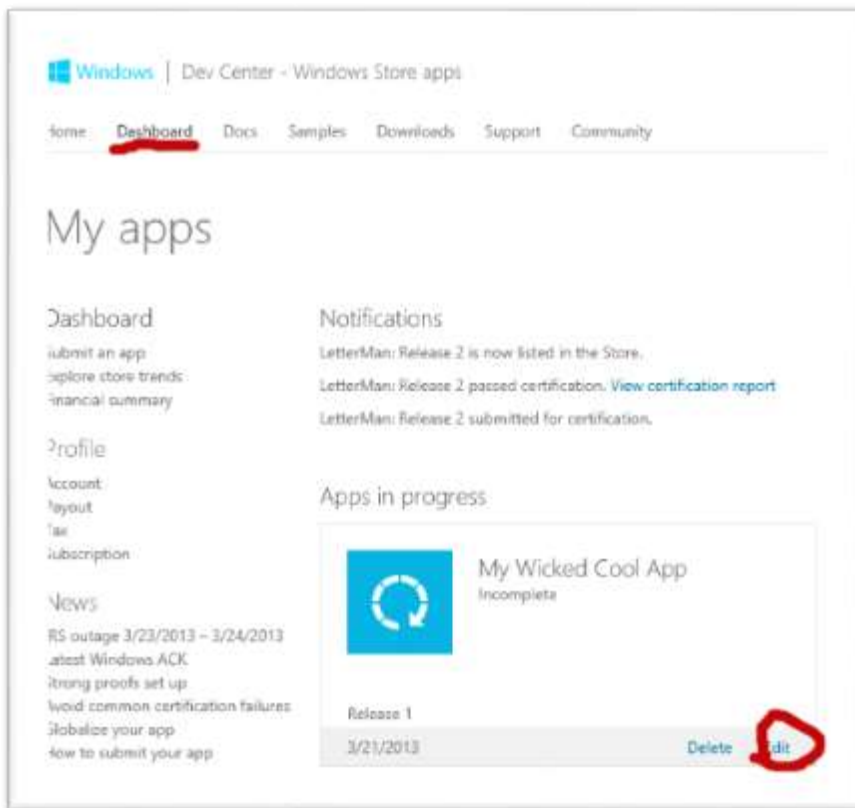
Exercise 5: Complete Your App Submission

At this point you have created the app package and it's time to complete the app submission process!

Task 1 – Complete the App Submission

In this task you'll complete submission by uploading the app package and completing the remaining details.

31. Return to the Windows Store developer dashboard.



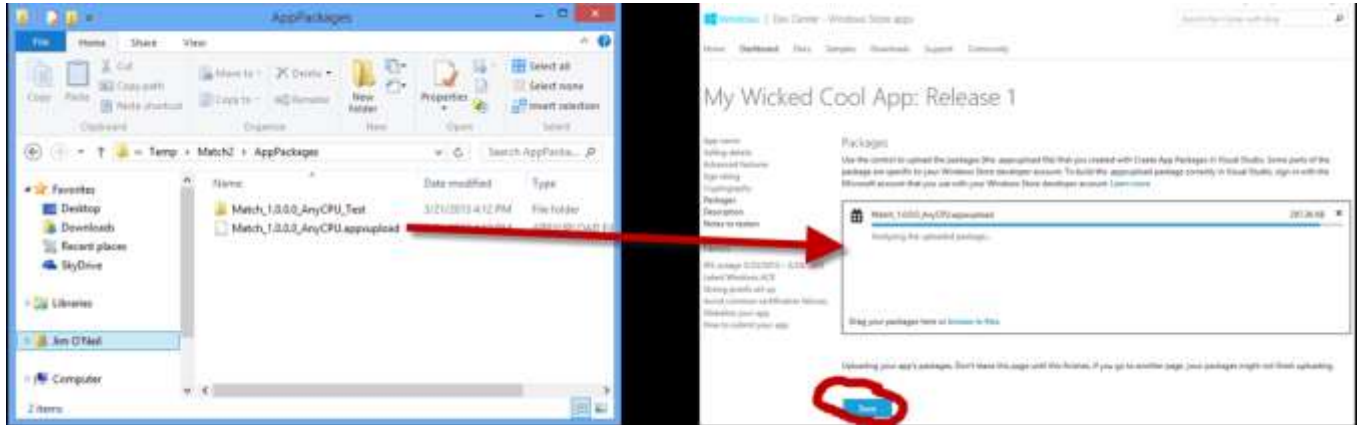
32. Packages



Packages

Upload your app to the Windows Store.
[Learn more](#)

33. Drag the *appxupload* file from the File Explorer windows over to the Packages field on the web page, and wait for the transfer to complete.



34. When that's done, press the *Save* button.

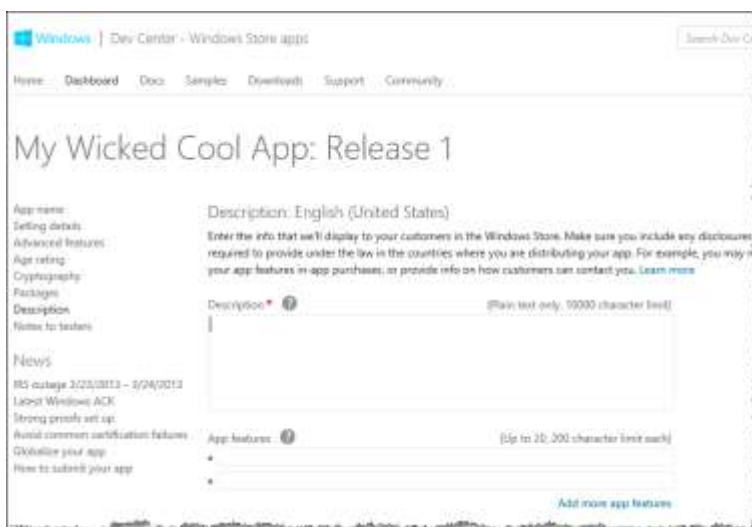
35. Description



Description

Briefly describe for your customers what your app does.

[Learn more](#)



You'll need to provide description information and graphics for your application listing on the Windows Store; see the table below for a description of these items as well as which of them are required.

Description (required)	A plain text description of your applications
App features	
Screenshots (at least one required)	Click the Add image link and browse to the screenshots you took earlier (in the <i>My Pictures/Windows Simulator</i>) directory. You will also enter some short text for each screen shot.
Description of update	Leave blank
Recommended hardware	Leave blank
Keywords	Provide search terms to help users find your app
Copyright and trademark info (required)	Use text like "Copyright 2013, <i>your-publisher-name</i> "

Promotional images	Leave blank
Website	Leave blank
Support contact info (required)	Enter your e-mail address
Privacy policy (required*)	Required since your app uses Internet connectivity. See item #4 in http://msdn.microsoft.com/en-us/library/windows/apps/hh694083.aspx for details.

36. Finally, press the Save button:



Privacy policy ? (2048 character limit)

Save

37. Notes to testers. Optional, but you must at least visit the page and click the Save button.



Notes to testers

Add notes about this release for the people who will review your app.

[Learn more](#)



My Wicked Cool App: Release 1

App name
Selling details
Advanced features
Age rating
Cryptography
Packages
Description
Notes to testers

News
IRS outage 3/23/2013 – 3/24/2013
Latest Windows ACK
Strong proofs set up
Avoid common certification failures
Globalize your app
How to submit your app

Notes to testers

Provide any info that helps the testers understand and use this app, in the Windows Store. [See example](#)

Here are some examples of info that can help us test your app.

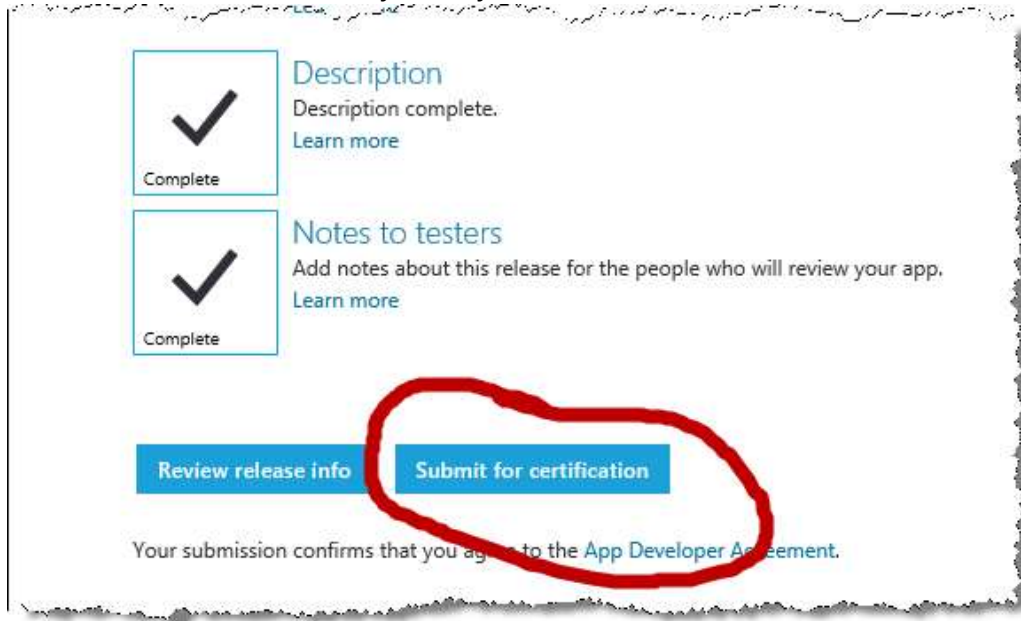
- If the app must log in to a service, provide the user name and password.
- If the app has features that might not be obvious to the testers, but incomplete will fail certification.
- If the app uses background audio, provide a test case that lets us reproduce.

Customers won't see the info that you enter on this page.

Instructions for testers:

Save

38. You're done! Press the *Submit for Certification* button!



That's it! You should quickly receive confirmation that your app is being reviewed. Now you just need to wait and you should hear (often in a couple of days, but maybe longer) about the results!

Summary

In this workbook, you took the steps to get your completed app submitted for certification in the Windows Store. Using your Windows Developer Account, you created an app, associated it with the project, created and uploaded an app package, and completed the app submission process.

Congratulations!