APIMASH Hands-on lab

Workbook 3: To the Windows Store

Lab version: 1.0.0

Last updated: 5/11/2013

developer & platform **evangelism**

CONTENTS

Overview

Welcome to the **APIMASH** program! This series will help you build useful Windows Store apps based on data and services provided by third parties via their own APIs. APIMASH features starter kits in XAML/C# and HTML/JavaScript that will help you quickly get started using these APIs, and into the Windows Store.

This is the last in a series of workbooks that will help you understand the starter kits, how they work, and how you can build your own Windows Store apps based on them:

- **Workbook 1 – Getting started with APIMASH**
  - Choosing and API & Starter Kit
  - Create an API Developer Account
  - Understanding the Starter Kit
  - Understanding the API Call
- **Workbook 2 – Controls, Data Binding, and Windows Store App Design**
  - Understanding data binding
  - Common Controls – GridView, ListView, and more
  - Windows store App features – Snapping, Sharing, Search, etc.
  - UI Design options
- **Workbook 3 (This workbook) – Getting Your App into the Windows Store**
  - Get a Windows Developer Account
  - Using the Dashboard
  - Submitting an App
  - Addressing Certification Issues

As you work though 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 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

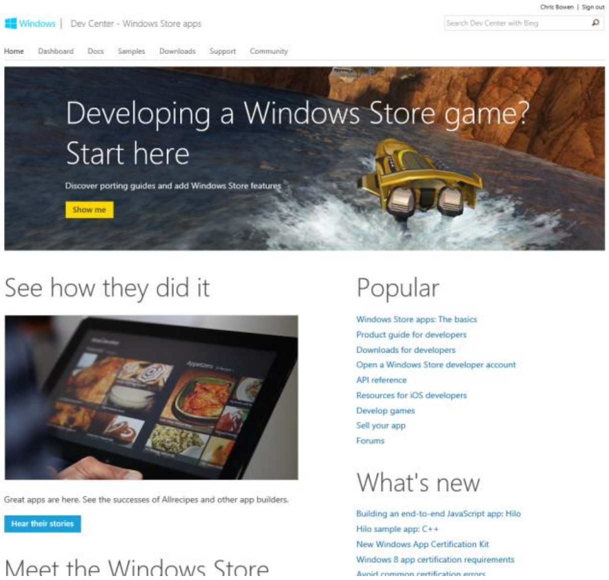
Estimated time to complete this workbook: **50 to 70 minutes**.

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


Chris Bowen | Sign out
Windows | Dev Center - Windows Store apps
Search Dev Center with Bing
Home
Dashboard
Docs
Samples
Downloads
Support
Community
Show me
Hear their stories
Popular
Windows Store apps: The basics
Product guide for developers
Downloads for developers
Open a Windows Store developer account
API reference
Resources for iOS developers
Develop games
Sell your app
Forums
What's new
Building an end-to-end JavaScript app: Hilo
Hilo sample app: C++
New Windows App Certification Kit
Windows 8 app certification requirements
Avoid common certification errors

Developing a Windows Store game? Start here

Discover porting guides and add Windows Store features

See how they did it

Great apps are here. See the successes of Allrecipes and other app builders.

Meet the Windows Store

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.

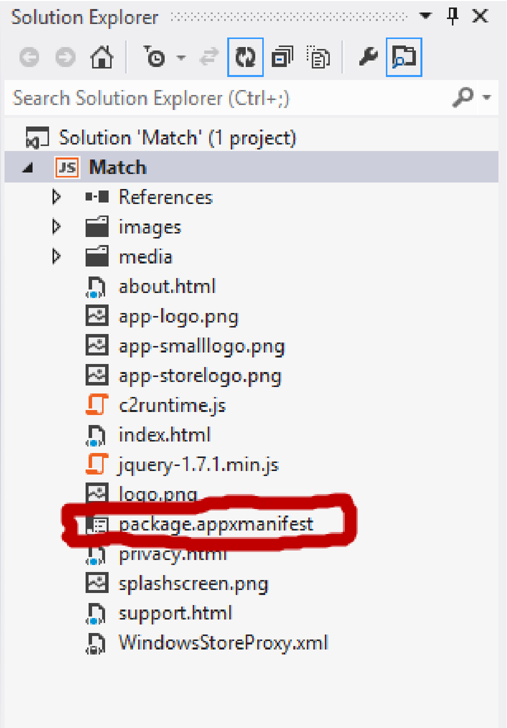
Exercise 2: Prepare Your App

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:

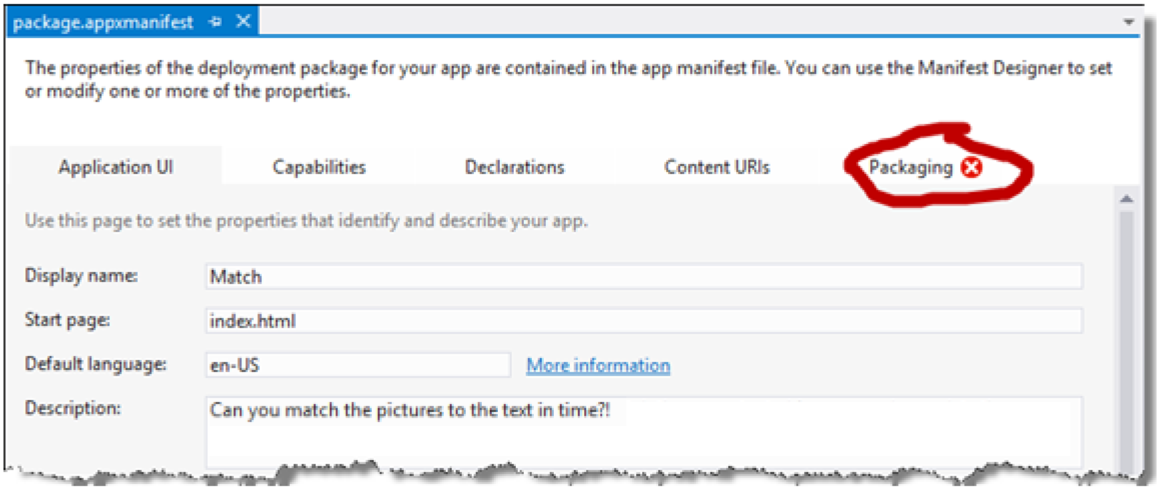


Solution Explorer

Search Solution Explorer (Ctrl+;)

- Solution 'Match' (1 project)
  - Match
    - References
    - images
    - media
    - about.html
    - app-logo.png
    - app-smalllogo.png
    - app-storelogo.png
    - c2runtime.js
    - index.html
    - jquery-1.7.1.min.js
    - logo.png
    - package.appxmanifest
    - privacy.html
    - splashscreen.png
    - support.html
    - WindowsStoreProxy.xml

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



package.appxmanifest

The properties of the deployment package for your app are contained in the app manifest file. You can use the Manifest Designer to set or modify one or more of the properties.

Use this page to set the properties that identify and describe your app.

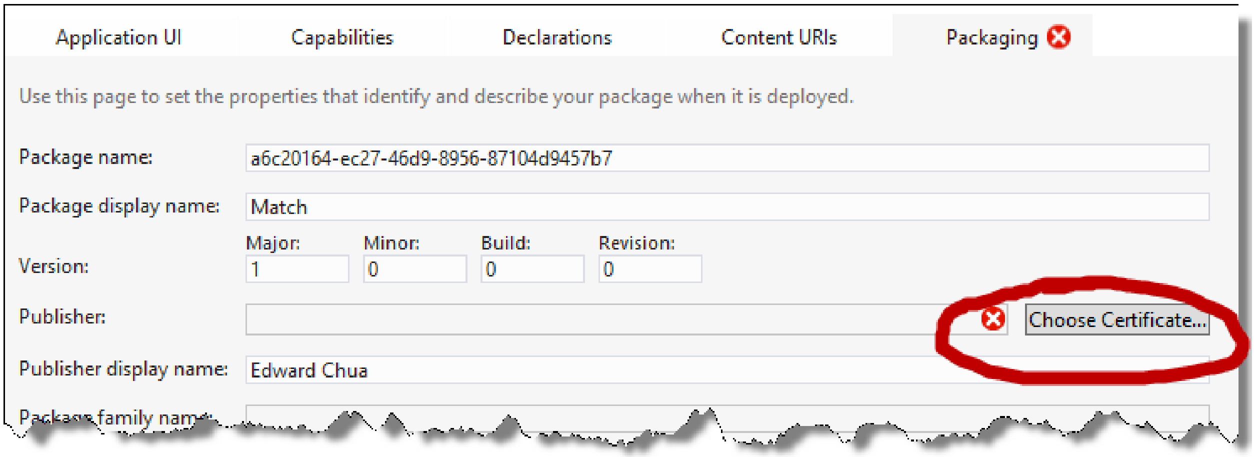
Display name: Match

Start page: index.html

Default language: en-US More information

Description: Can you match the pictures to the text in time?!

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



Use this page to set the properties that identify and describe your package when it is deployed.

Package name: a6c20164-ec27-46d9-8956-87104d9457b7

Package display name: Match

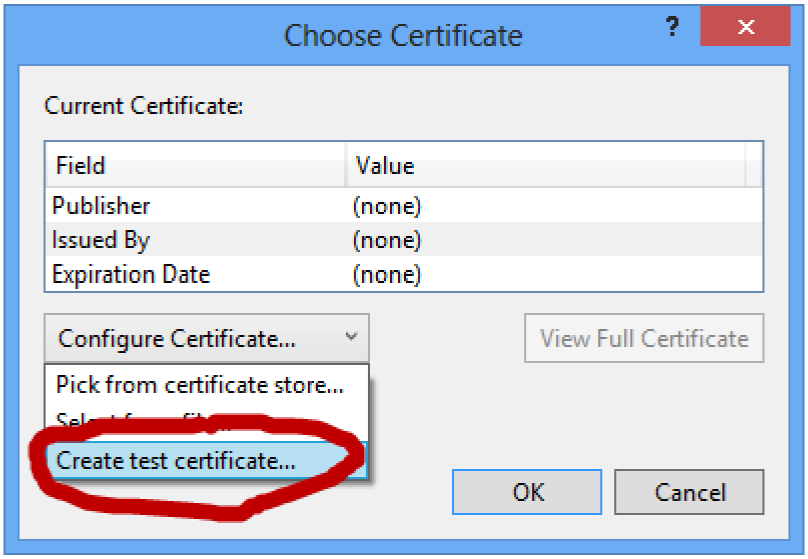
Version: Major: 1 Minor: 0 Build: 0 Revision: 0

Publisher:

Publisher display name: Edward Chua

Package family name:

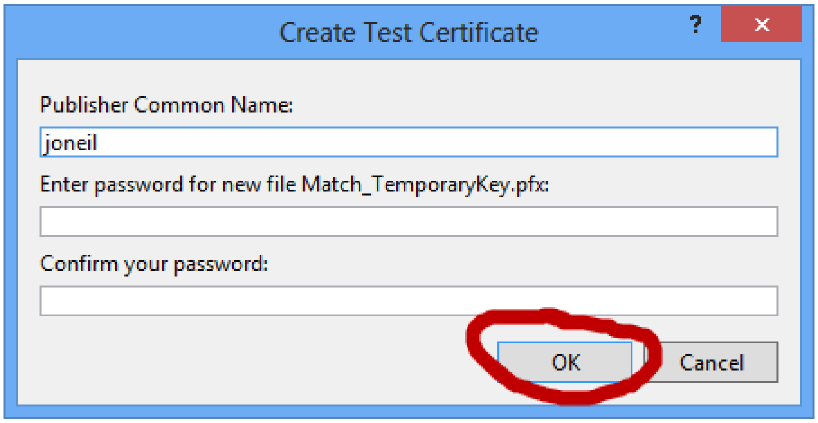
4. Then pick *Create Test Certificate*, and press *OK*.



Current Certificate:

Field	Value
Publisher	(none)
Issued By	(none)
Expiration Date	(none)

5. Press OK on the next dialog (you don’t need to enter anything in the text boxes)

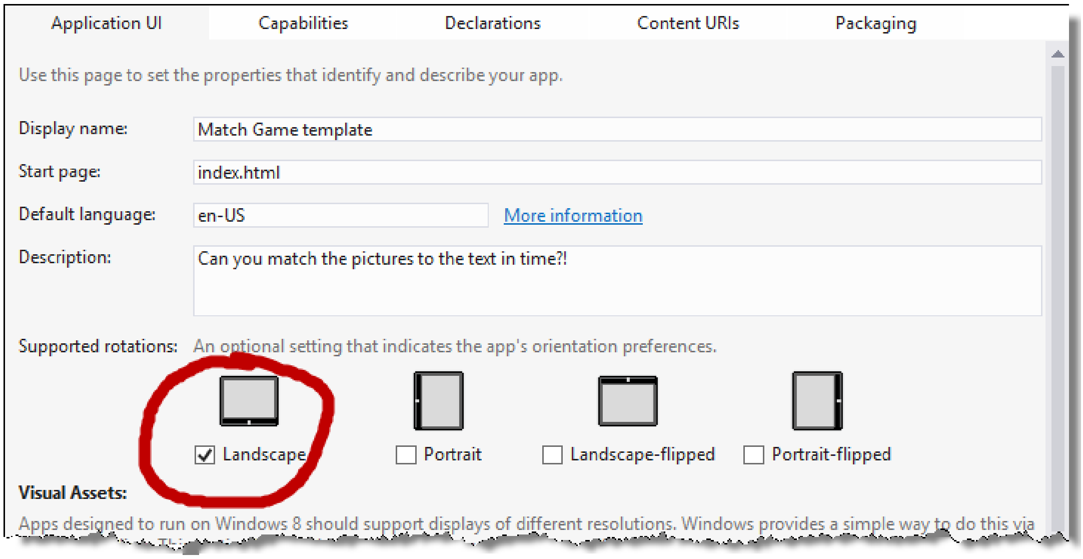


Publisher Common Name: joneil

Enter password for new file Match_TemporaryKey.pfx:

Confirm your password:

6. 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.



Use this page to set the properties that identify and describe your app.

Display name: Match Game template

Start page: index.html

Default language: en-US More information

Description: Can you match the pictures to the text in time?!

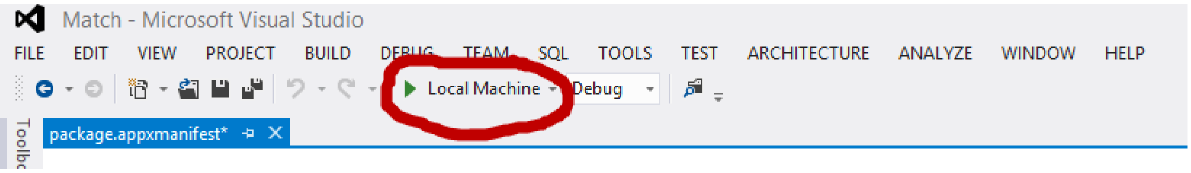
Supported rotations: An optional setting that indicates the app's orientation preferences.

☑ Landscape ☐ Portrait ☐ Landscape-flipped ☐ Portrait-flipped

Visual Assets:

Apps designed to run on Windows 8 should support displays of different resolutions. Windows provides a simple way to do this via

7. 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.



8. 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.

1. 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

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

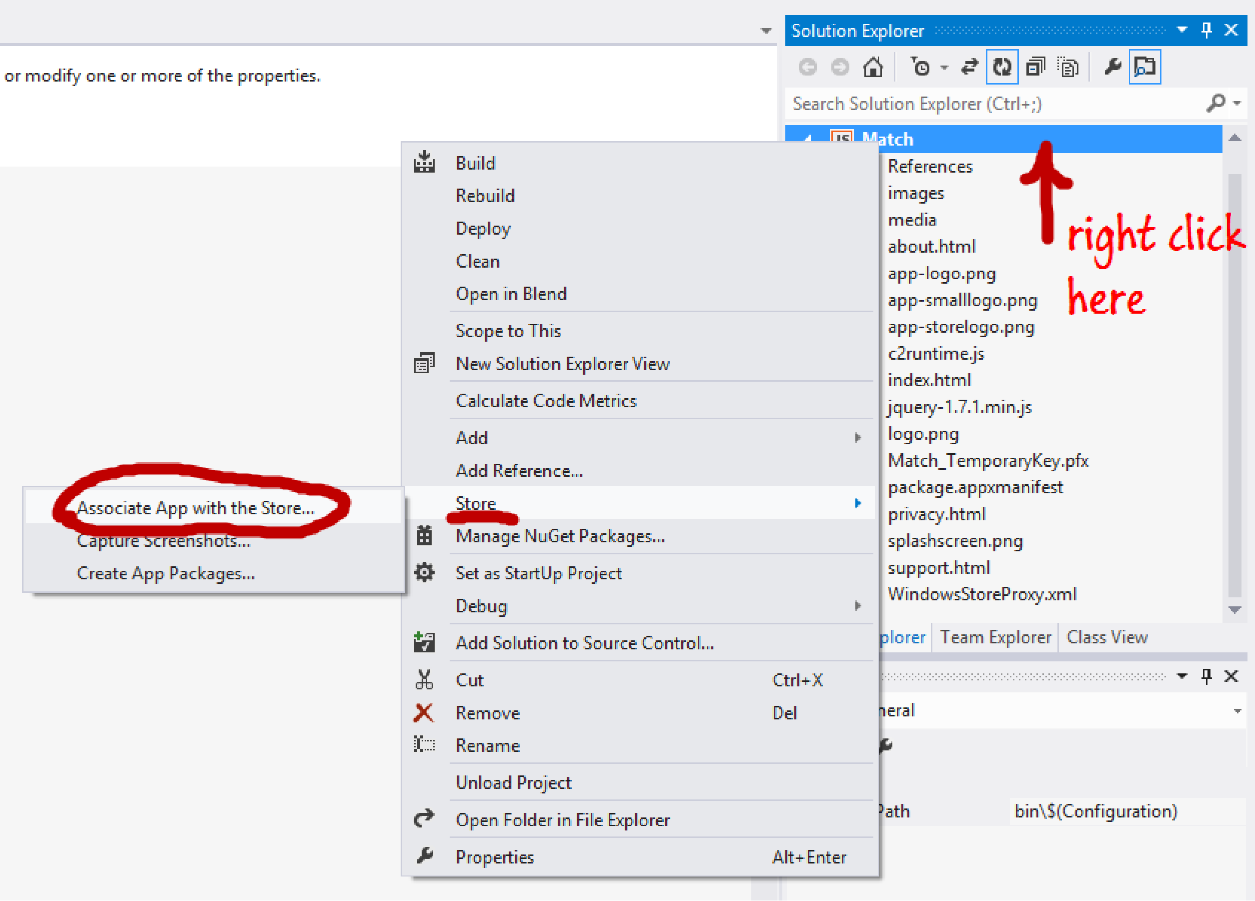
Exercise 3: Beginning the Store Submission

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

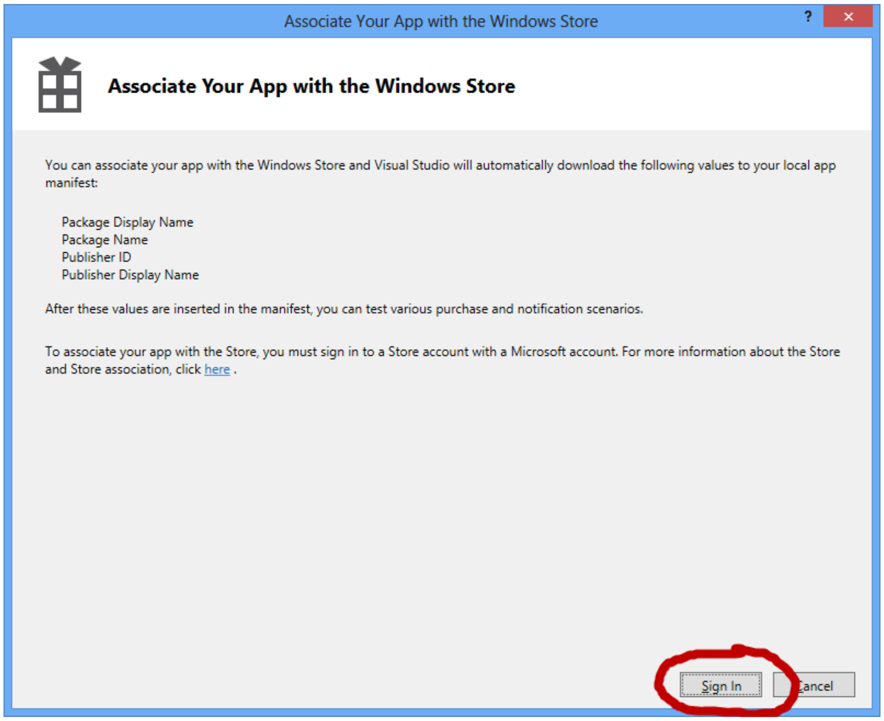
Task 1 – Associating Your App with the Store

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



Associate Your App with the Windows Store

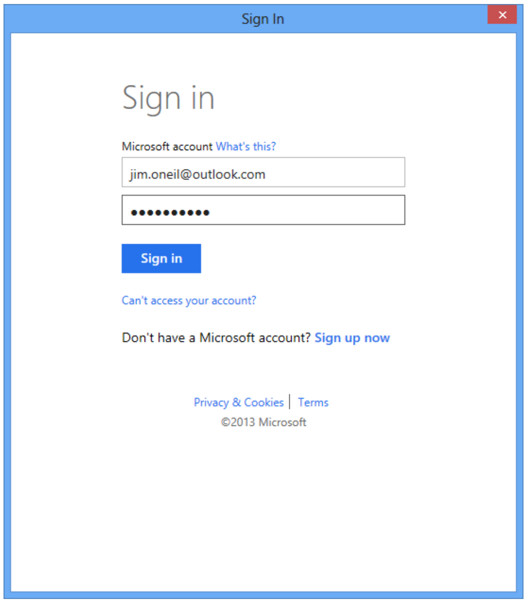
You can associate your app with the Windows Store and Visual Studio will automatically download the following values to your local app manifest:

Package Display Name
Package Name
Publisher ID
Publisher Display Name

After these values are inserted in the manifest, you can test various purchase and notification scenarios.

To associate your app with the Store, you must sign in to a Store account with a Microsoft account. For more information about the Store and Store association, click here .

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



Sign In

Sign in

Microsoft account What's this?

jim.oneil@outlook.com

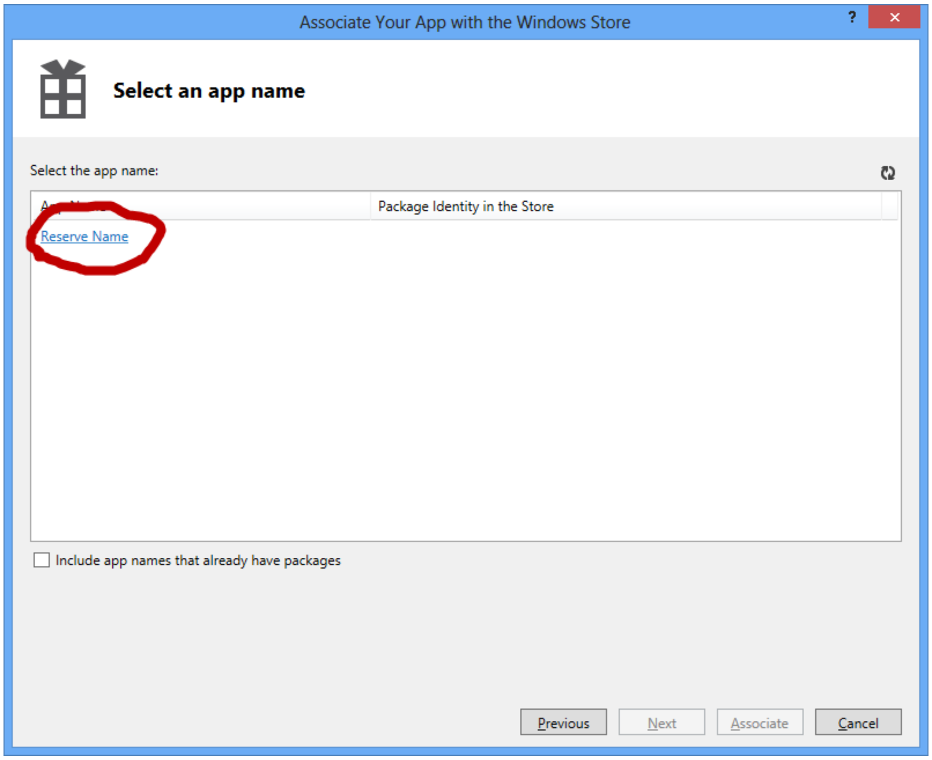
••••••••••

Can't access your account?

Don't have a Microsoft account? Sign up now

Privacy & Cookies | Terms
©2013 Microsoft

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



Associate Your App with the Windows Store

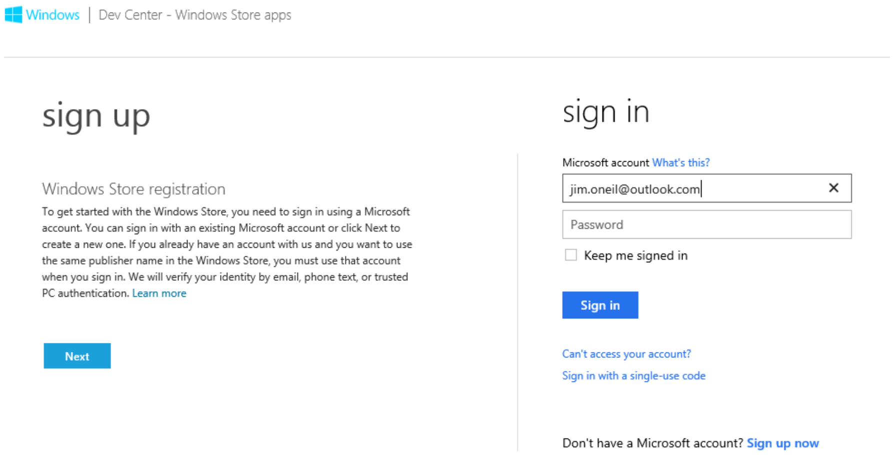
Select an app name

Select the app name:

App Name	Package Identity in the Store
Reserve Name	

□ Include app names that already have packages

5. You'll be asked to sign in again



Windows | Dev Center - Windows Store apps

sign up

Windows Store registration

To get started with the Windows Store, you need to sign in using a Microsoft account. You can sign in with an existing Microsoft account or click Next to create a new one. If you already have an account with us and you want to use the same publisher name in the Windows Store, you must use that account when you sign in. We will verify your identity by email, phone text, or trusted PC authentication. Learn more

sign in

Microsoft account What's this?

jim.oneil@outlook.com

Password

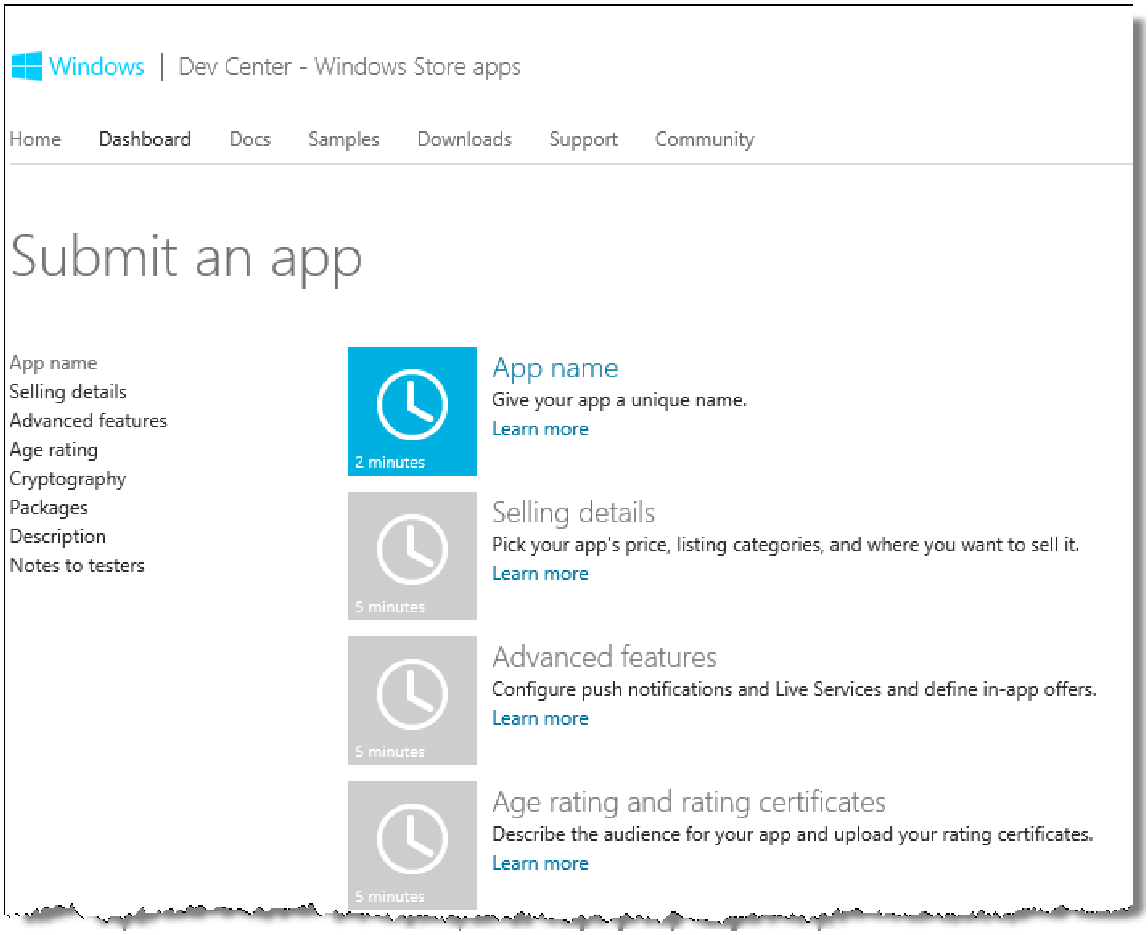
□ Keep me signed in

Can't access your account?

Sign in with a single-use code

Don't have a Microsoft account? Sign up now

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



Submit an app

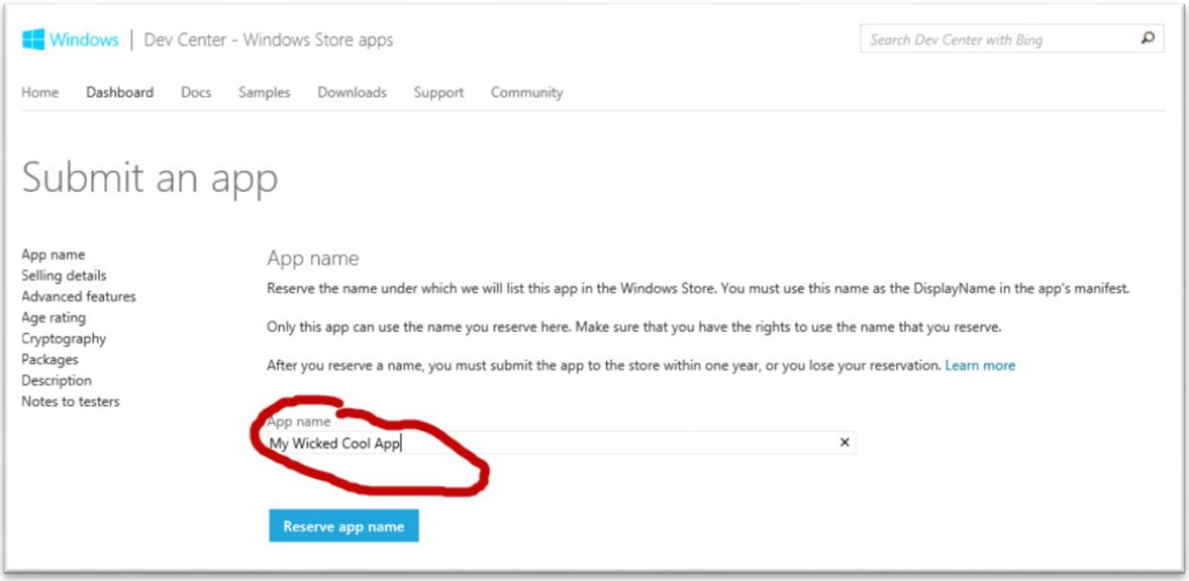
App name
Give your app a unique name.
Learn more

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

Advanced features
Configure push notifications and Live Services and define in-app offers.
Learn more

Age rating and rating certificates
Describe the audience for your app and upload your rating certificates.
Learn more

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.



Submit an app

App name

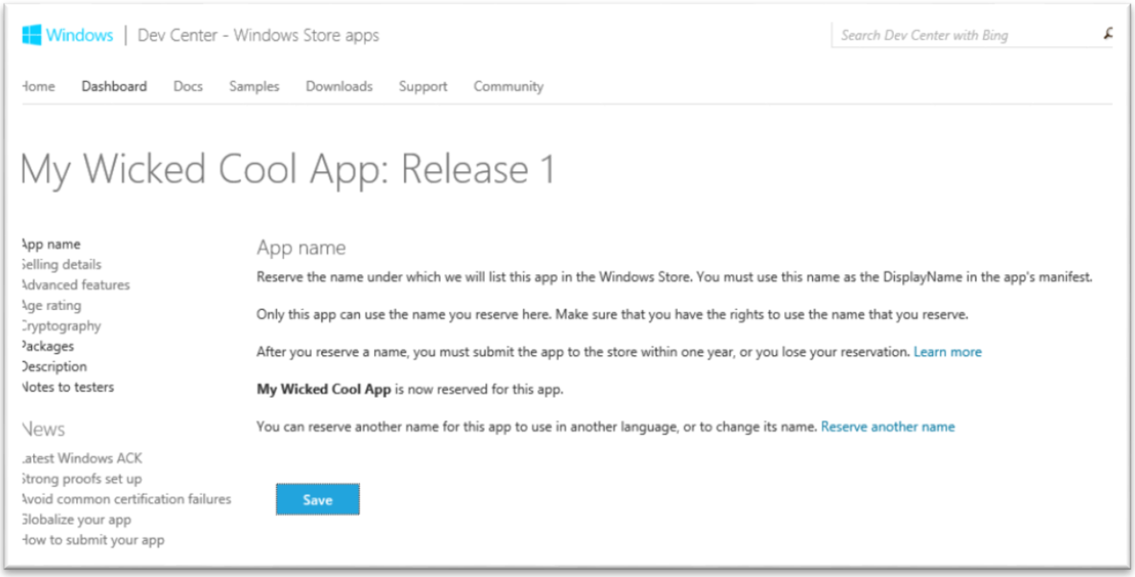
Reserve the name under which we will list this app in the Windows Store. You must use this name as the DisplayName in the app's manifest.

Only this app can use the name you reserve here. Make sure that you have the rights to use the name that you reserve.

After you reserve a name, you must submit the app to the store within one year, or you lose your reservation. Learn more

App name: My Wicked Cool App

8. Press the *Save* button



My Wicked Cool App: Release 1

App name

Reserve the name under which we will list this app in the Windows Store. You must use this name as the DisplayName in the app's manifest.

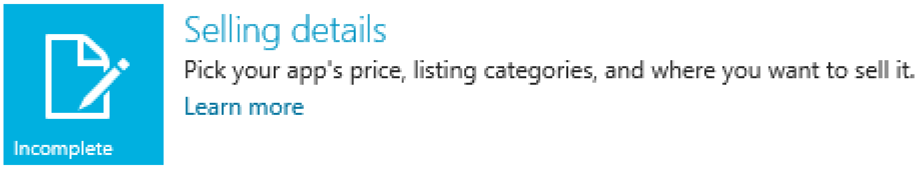
Only this app can use the name you reserve here. Make sure that you have the rights to use the name that you reserve.

After you reserve a name, you must submit the app to the store within one year, or you lose your reservation. Learn more

My Wicked Cool App is now reserved for this app.

You can reserve another name for this app to use in another language, or to change its name. Reserve another name

9. Continue by clicking **Selling details**

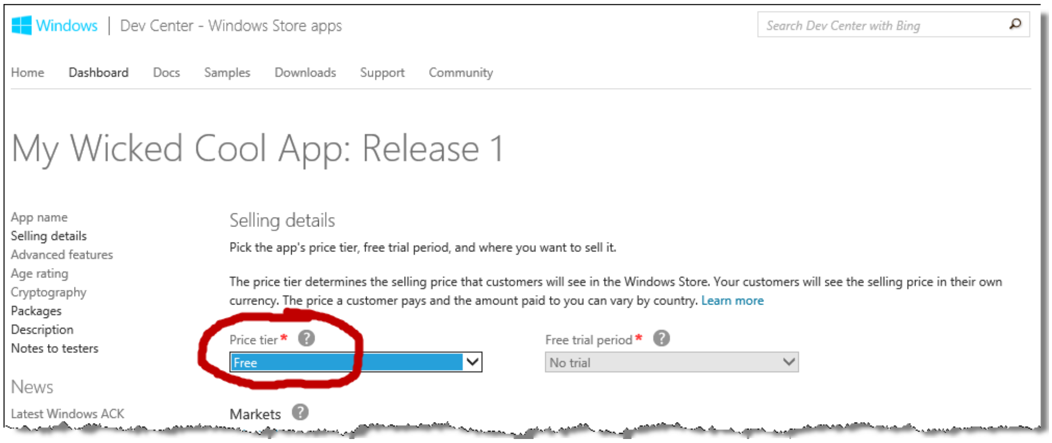


Selling details

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.



My Wicked Cool App: Release 1

Selling details

Pick the app's price tier, free trial period, and where you want to sell it.

The price tier determines the selling price that customers will see in the Windows Store. Your customers will see the selling price in their own currency. The price a customer pays and the amount paid to you can vary by country. Learn more

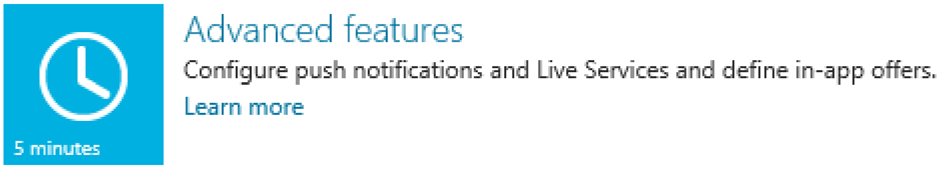
Price tier*: Free

Free trial period*: No trial

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 *Release my app as soon as it passes certification.*
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 *Available to all systems.*
Accessibility	Leave this **unchecked**

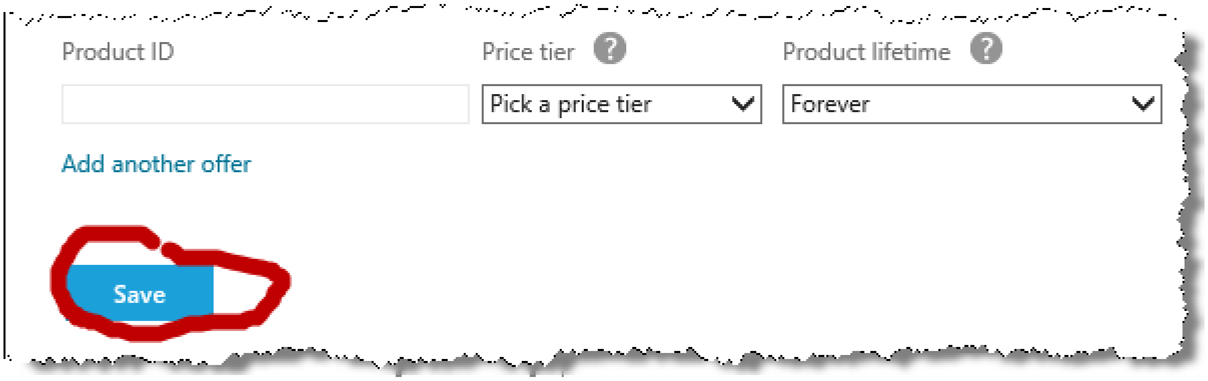
12. Continue with **Advanced features,** do nothing except press the *Save* button



Advanced features

Configure push notifications and Live Services and define in-app offers.

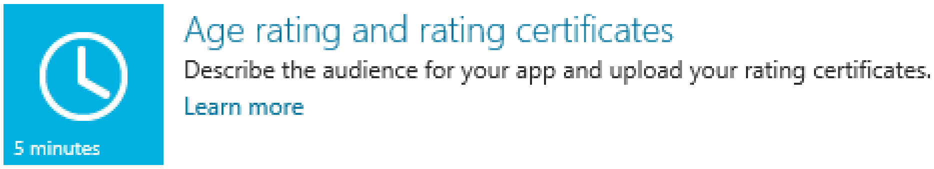
Learn more



Product ID	Price tier	Product lifetime
	Pick a price tier	Forever

Add another offer

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

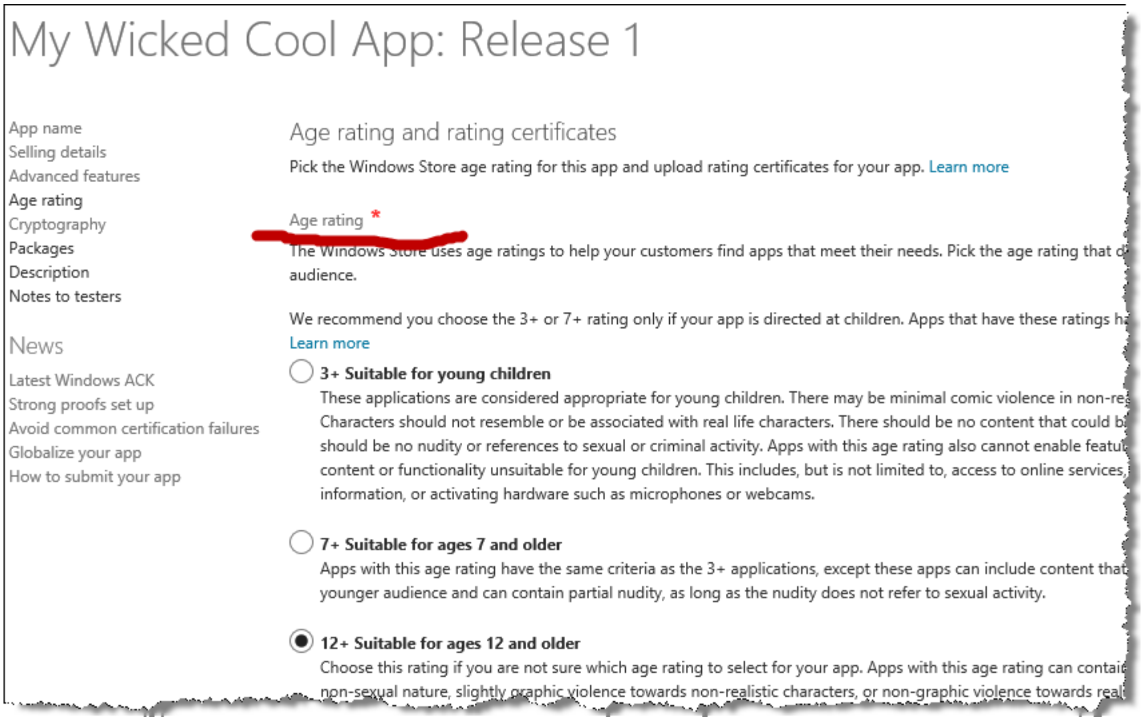


Age rating and rating certificates

Describe the audience for your app and upload your rating certificates.

Learn more

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

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 d
audience.

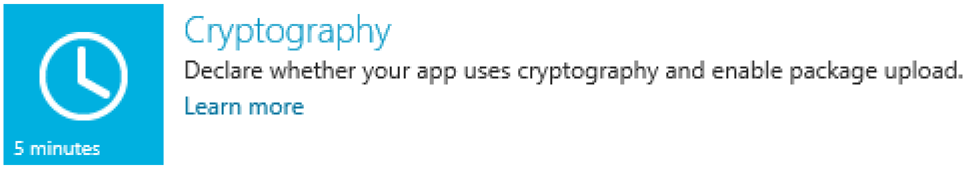
We recommend you choose the 3+ or 7+ rating only if your app is directed at children. Apps that have these ratings h
Learn more

○ **3+ Suitable for young children**
These applications are considered appropriate for young children. There may be minimal comic violence in non-re
Characters should not resemble or be associated with real life characters. There should be no content that could b
should be no nudity or references to sexual or criminal activity. Apps with this age rating also cannot enable featu
content or functionality unsuitable for young children. This includes, but is not limited to, access to online services,
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
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 contai
non-sexual nature, slightly graphic violence towards non-realistic characters, or non-graphic violence towards real

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

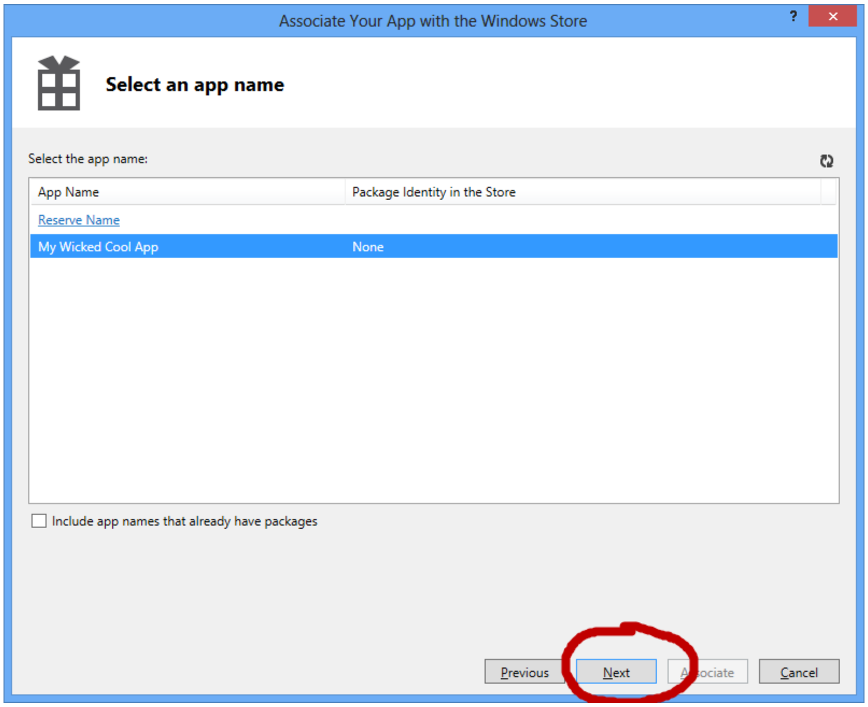
At this point you have created most of the application profile in your Store developer account! Next you'll create the app package for upload you'll go back to Visual Studio to create a package to upload your app to this new application profile.

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.



Associate Your App with the Windows Store

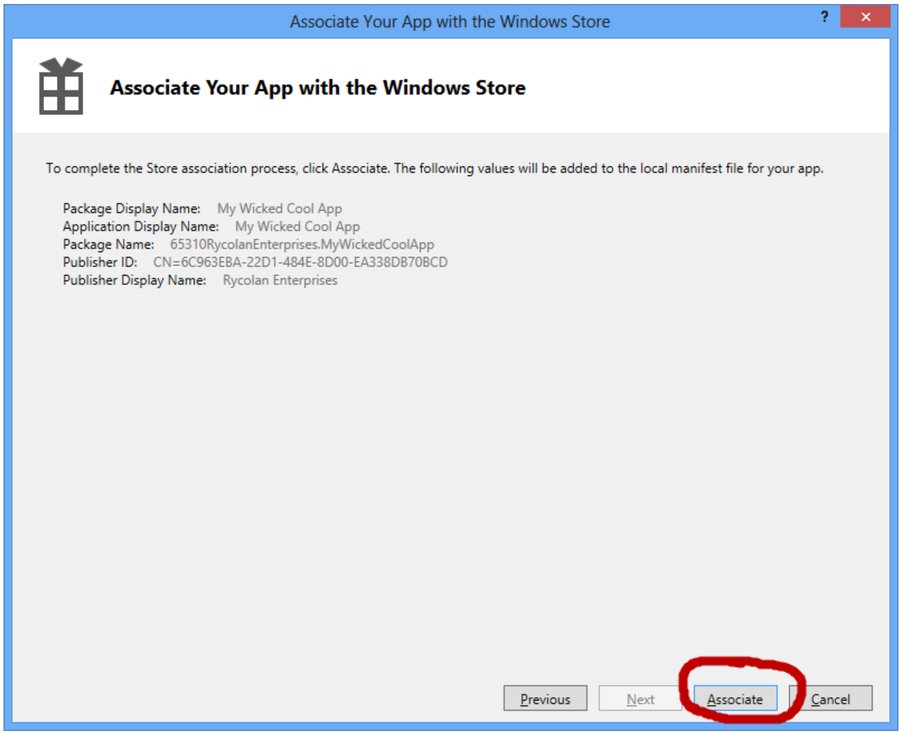
Select an app name

Select the app name:

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

□ Include app names that already have packages

19. Now press the *Associate* button:

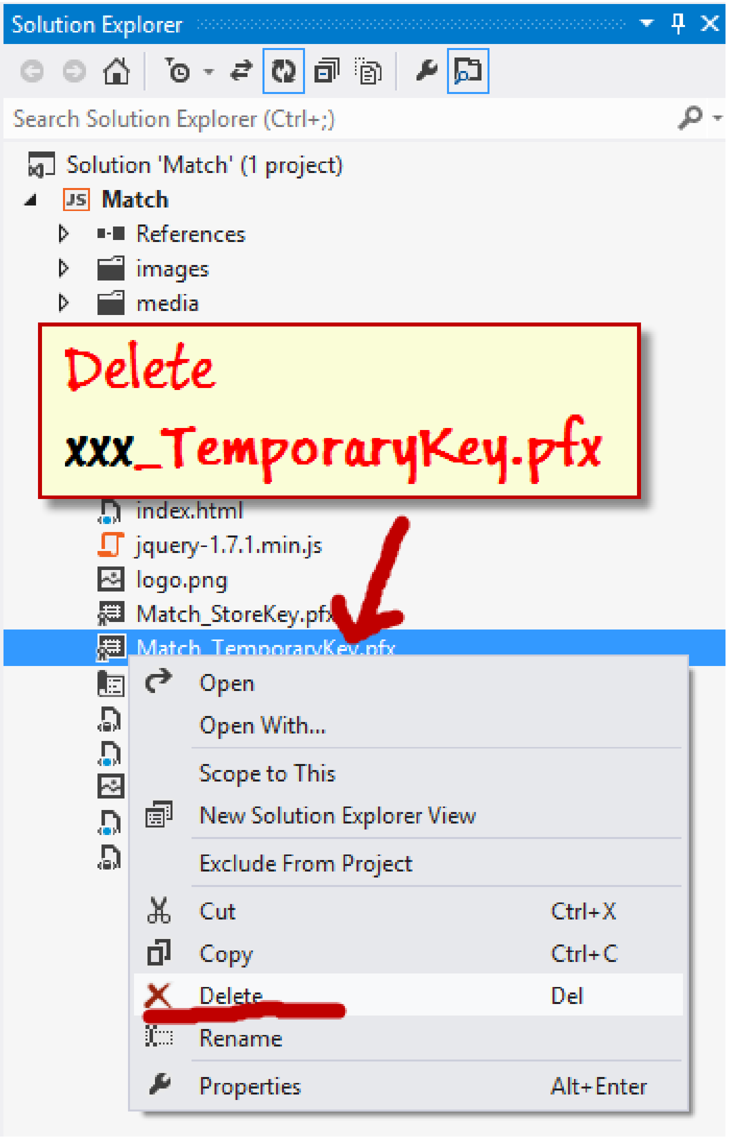


Associate Your App with the Windows Store

To complete the Store association process, click Associate. The following values will be added to the local manifest file for your app.

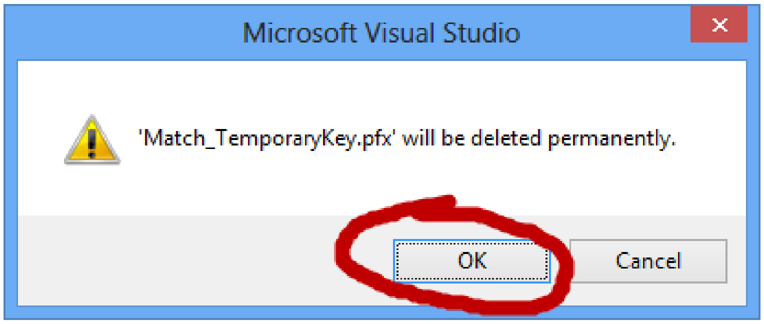
Package Display Name:	My Wicked Cool App
Application Display Name:	My Wicked Cool App
Package Name:	65310RycolanEnterprises.MyWickedCoolApp
Publisher ID:	CN=6C963EBA-22D1-484E-8D00-EA338DB70BCD
Publisher Display Name:	Rycolan Enterprises

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.



Delete xxx_TemporaryKey.pfx

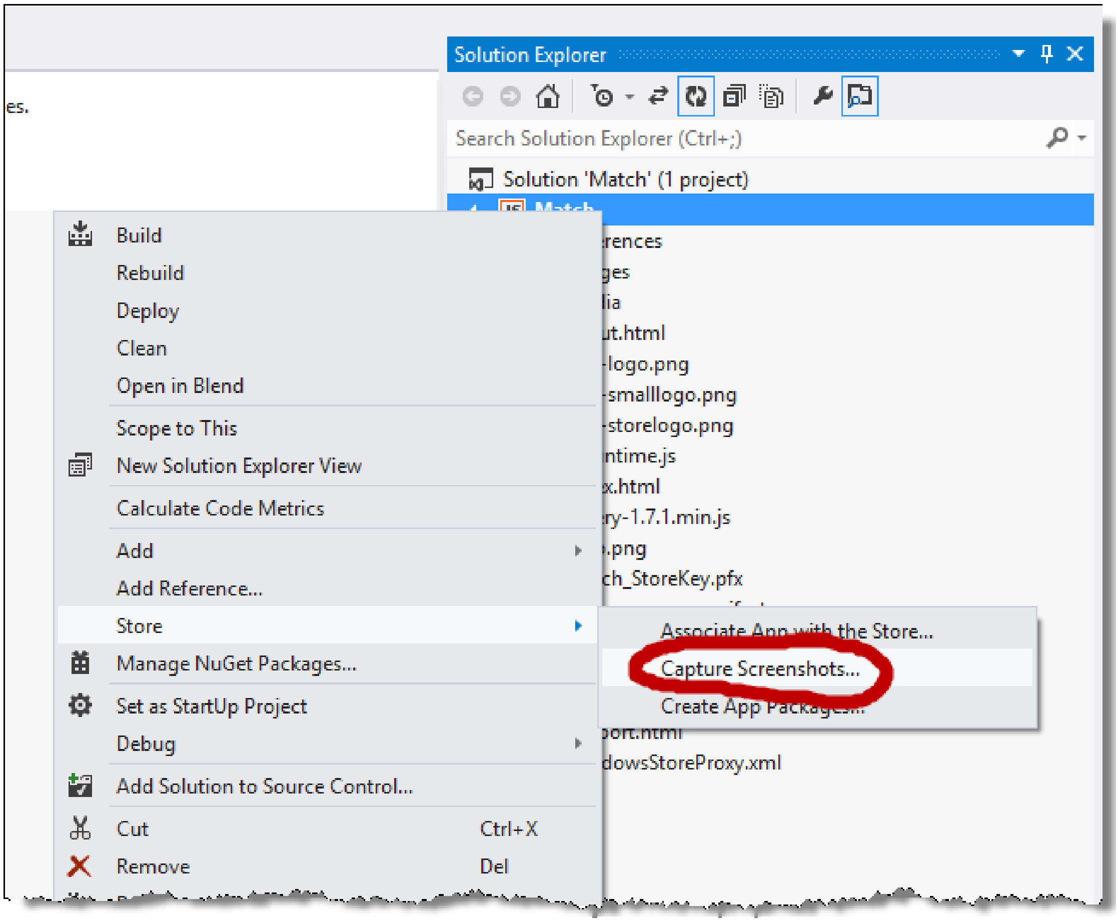
21. Press OK when prompted to confirm you want to delete permanently.



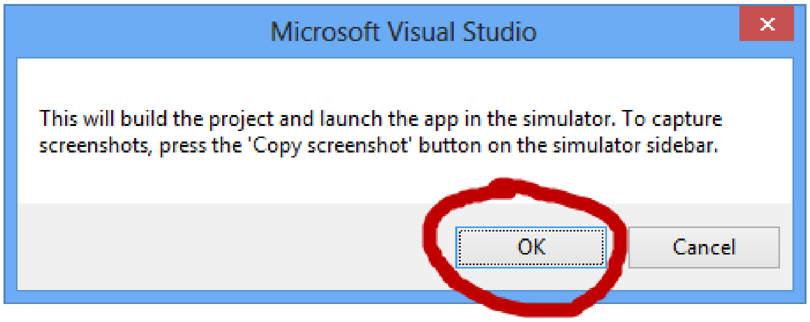
Microsoft Visual Studio

'Match_TemporaryKey.pfx' will be deleted 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:

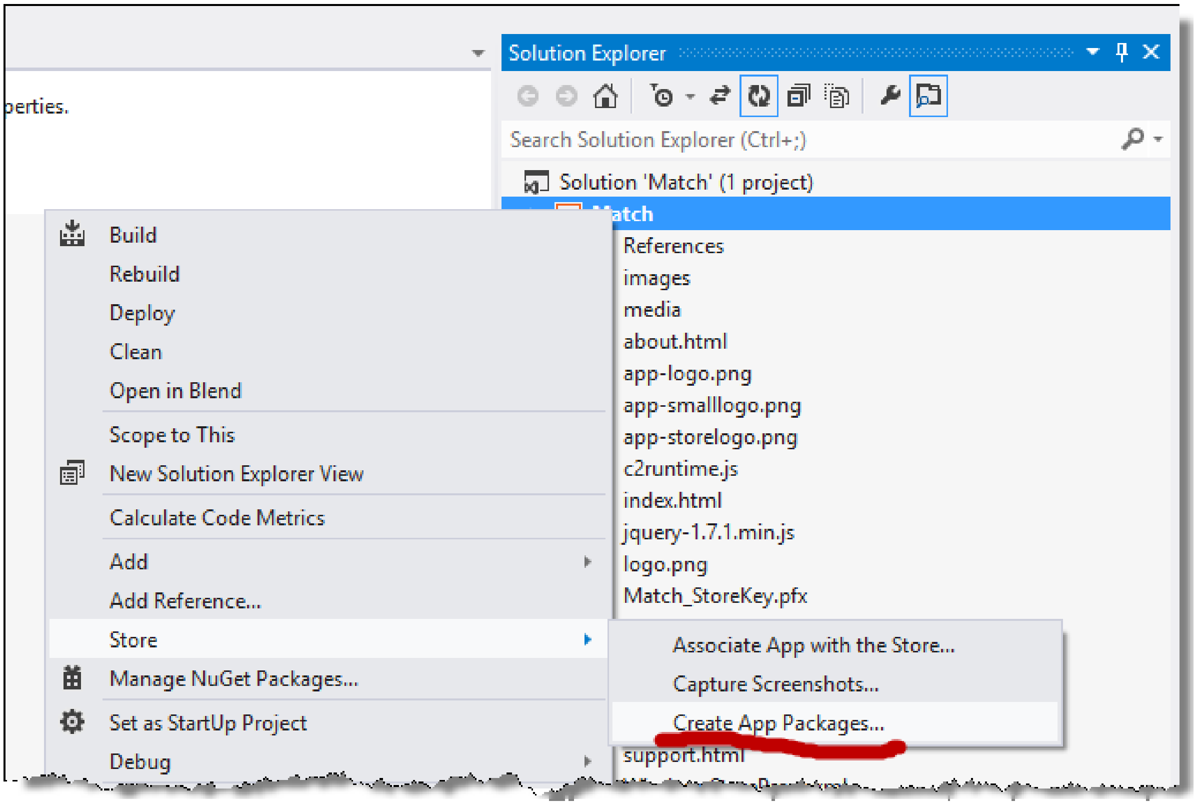

Microsoft Visual Studio

This will build the project and launch the app in the simulator. To capture screenshots, press the 'Copy screenshot' button on the simulator sidebar.

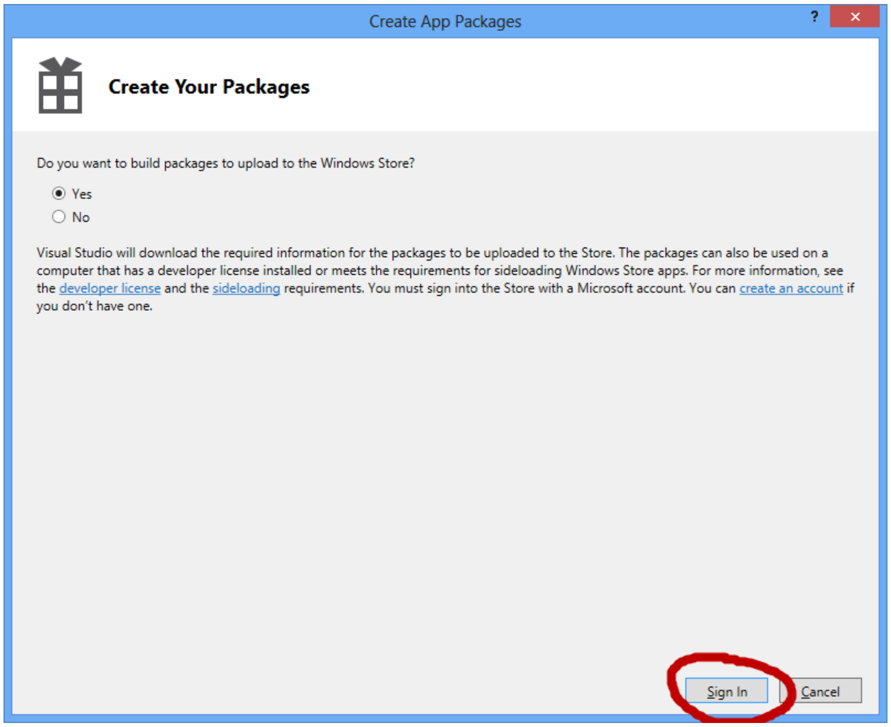
The Simulator should automatically start and begin running your app. Use the app the show features you want to highlight, then click the Camera button on the right-hand toolbar to take one or more screenshots.

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.



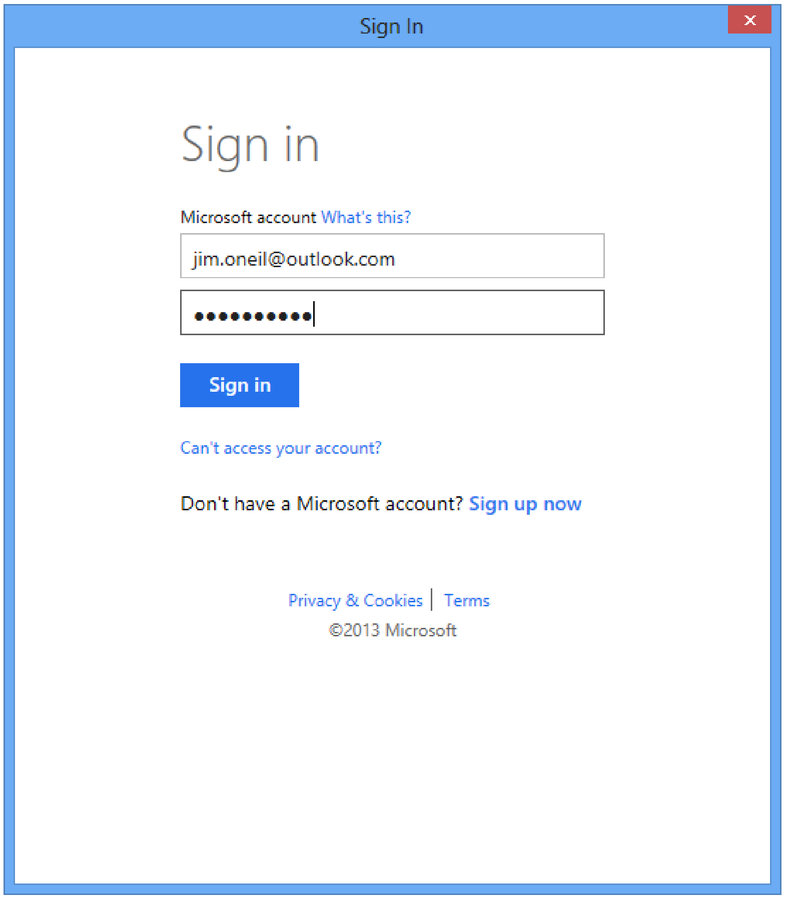
Create Your Packages

Do you want to build packages to upload to the Windows Store?

● Yes
○ No

Visual Studio will download the required information for the packages to be uploaded to the Store. The packages can also be used on a computer that has a developer license installed or meets the requirements for sideloading Windows Store apps. For more information, see the developer license and the sideloading requirements. You must sign into the Store with a Microsoft account. You can create an account if you don't have one.

26. Sign into your developer account again.



Microsoft account What's this?

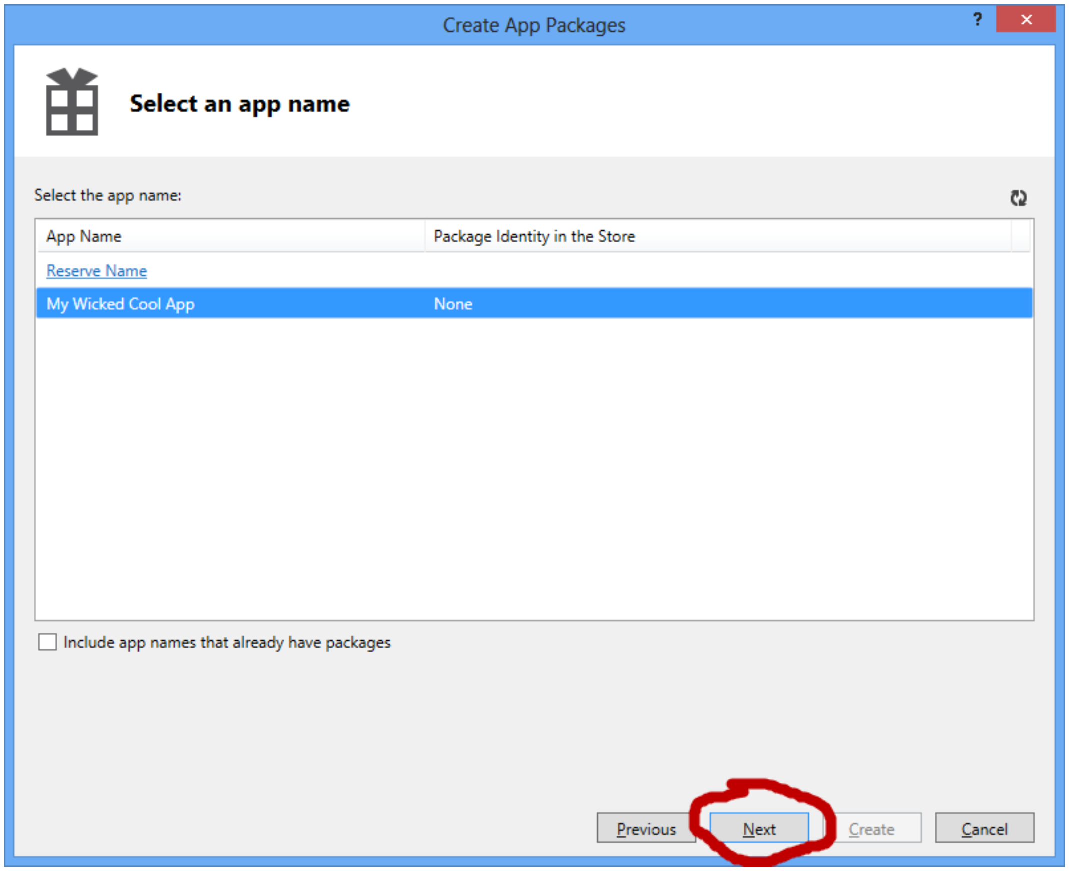
jim.oneil@outlook.com

●●●●●●●●●●

Don't have a Microsoft account? Sign up now

©2013 Microsoft

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

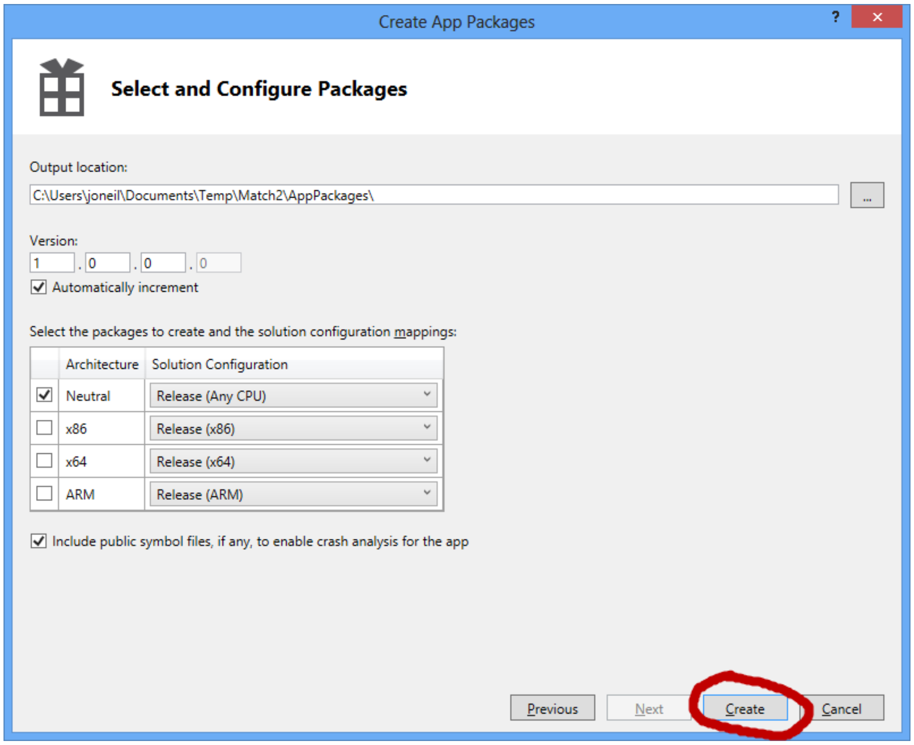


Select the app name:

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

□ Include app names that already have packages

28. On the screen that follows, press Create



Output location:
C:\Users\joneil\Documents\Temp\Match2\AppPackages\

Version:
1 . 0 . 0 . 0

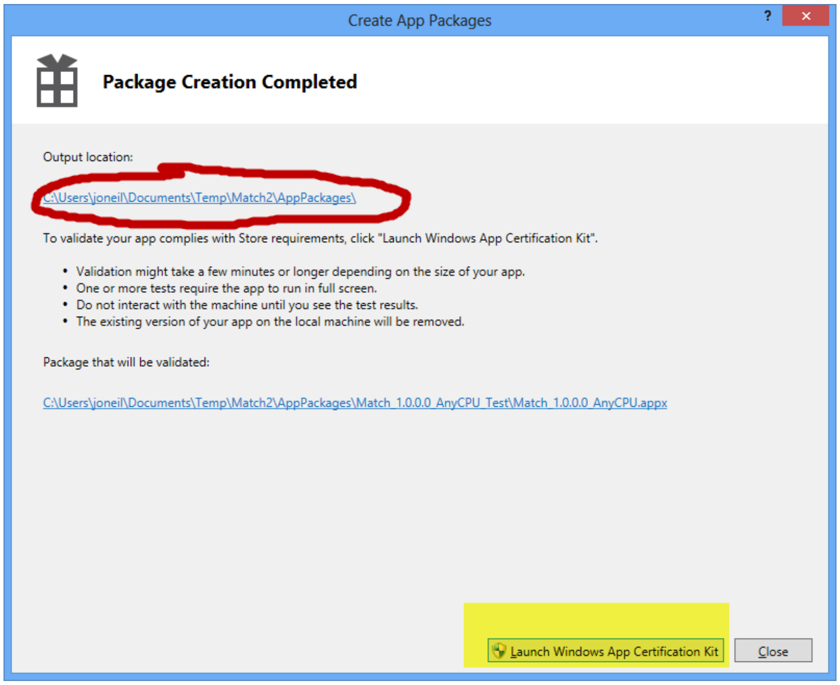
☑ Automatically increment

Select the packages to create and the solution configuration mappings:

	Architecture	Solution Configuration
☑	Neutral	Release (Any CPU)
□	x86	Release (x86)
□	x64	Release (x64)
□	ARM	Release (ARM)

☑ Include public symbol files, if any, to enable crash analysis for the app

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



Package Creation Completed

Output location:

C:\Users\joneil\Documents\Temp\Match2\AppPackages\

To validate your app complies with Store requirements, click "Launch Windows App Certification Kit".

- Validation might take a few minutes or longer depending on the size of your app.
- One or more tests require the app to run in full screen.
- Do not interact with the machine until you see the test results.
- The existing version of your app on the local machine will be removed.

Package that will be validated:

C:\Users\joneil\Documents\Temp\Match2\AppPackages\Match_1.0.0.0_AnyCPU_Test\Match_1.0.0.0_AnyCPU.appx

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

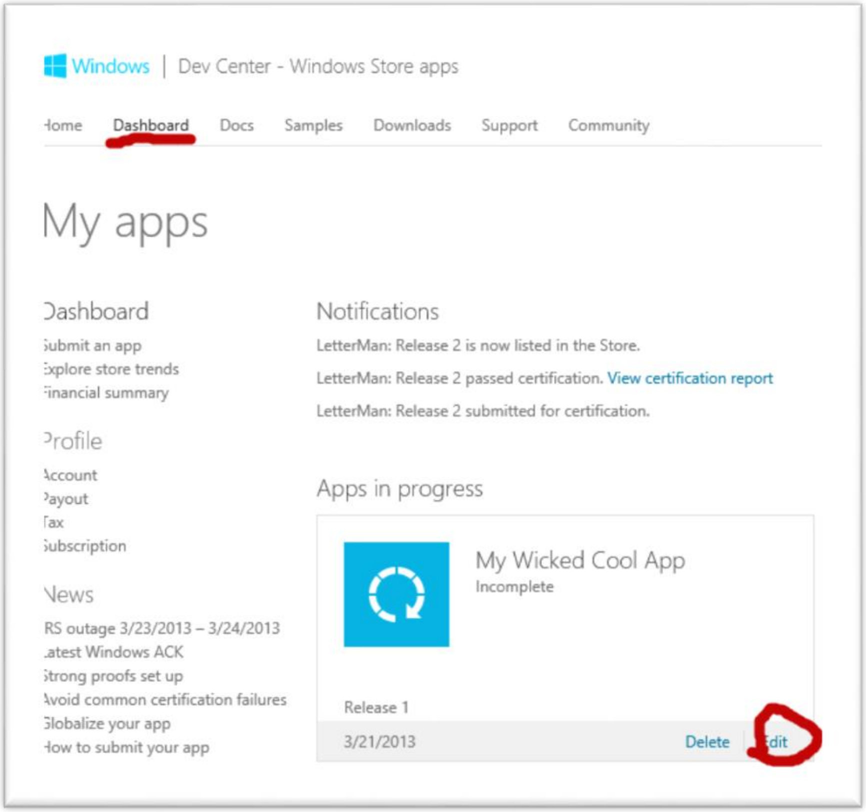
Exercise 5: Complete Your App Submission

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.



My apps

ashboard

ubmit an app
xplore store trends
inancial summary

rofile

ccount
ayout
ax
ubscription

ews

RS outage 3/23/2013 – 3/24/2013
atest Windows ACK
trong proofs set up
void common certification failures
lobalize your app
ow to submit your app

Notifications

LetterMan: Release 2 is now listed in the Store.

LetterMan: Release 2 passed certification. View certification report

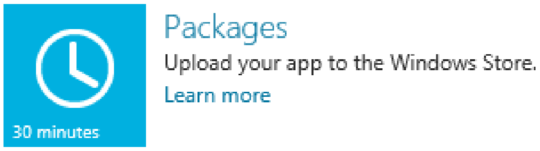
LetterMan: Release 2 submitted for certification.

Apps in progress

My Wicked Cool App
Incomplete

Release 1
3/21/2013

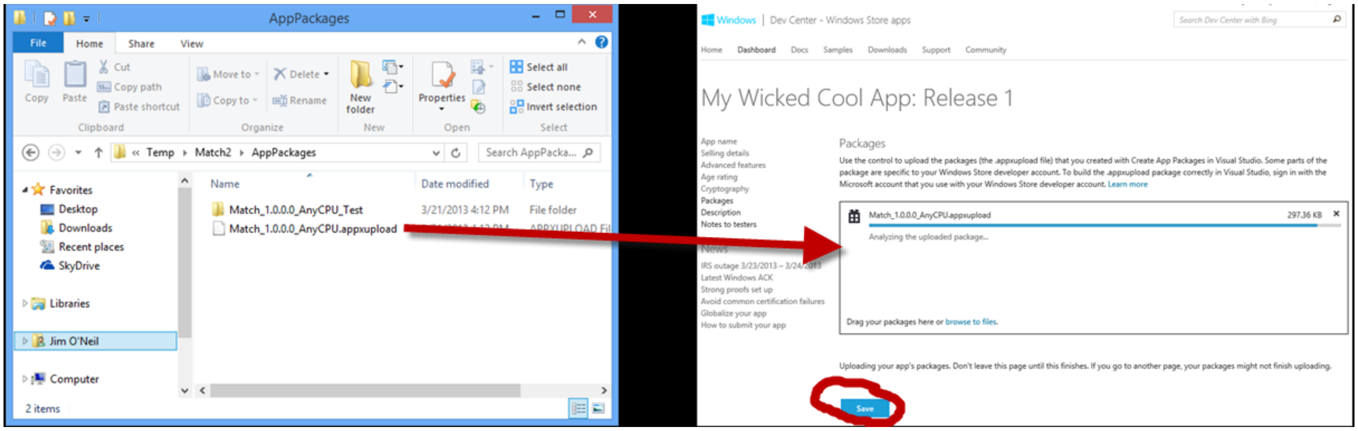
32. Packages



Packages

Upload your app to the Windows Store.

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.



Match_1.0.0.0_AnyCPU_Test 3/21/2013 4:12 PM File folder

Match_1.0.0.0_AnyCPU.appxupload

My Wicked Cool App: Release 1

Packages

Use the control to upload the packages (the .appxupload file) that you created with Create App Packages in Visual Studio. Some parts of the package are specific to your Windows Store developer account. To build the .appxupload package correctly in Visual Studio, sign in with the Microsoft account that you use with your Windows Store developer account. Learn more

Match_1.0.0.0_AnyCPU.appxupload 297.36 KB

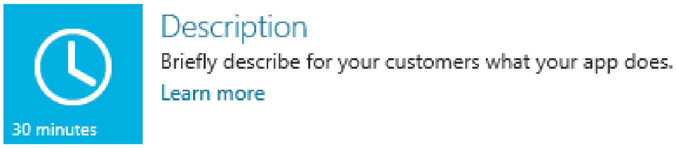
Analyzing the uploaded package...

Drag your packages here or browse to files.

Uploading your app's packages. Don't leave this page until this finishes. If you go to another page, your packages might not finish uploading.

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

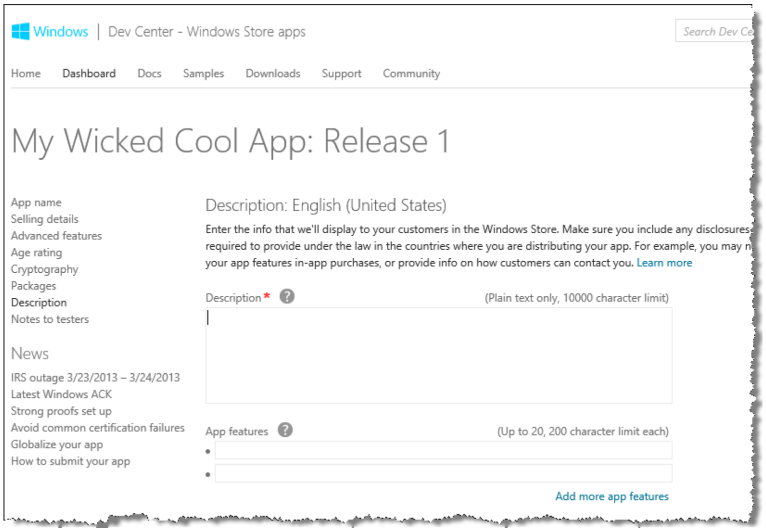
35. Description



Description

Briefly describe for your customers what your app does.

Learn more



My Wicked Cool App: Release 1

Description: English (United States)

Enter the info that we'll display to your customers in the Windows Store. Make sure you include any disclosures required to provide under the law in the countries where you are distributing your app. For example, you may n your app features in-app purchases, or provide info on how customers can contact you. Learn more

Description * (Plain text only, 10000 character limit)

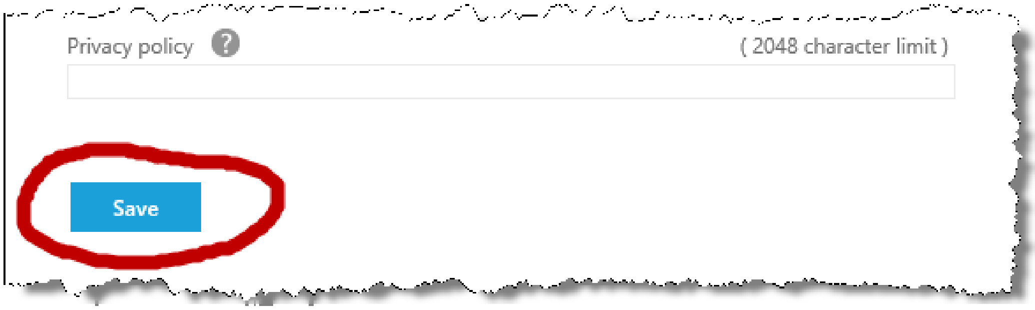
App features (Up to 20, 200 character limit each)

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.

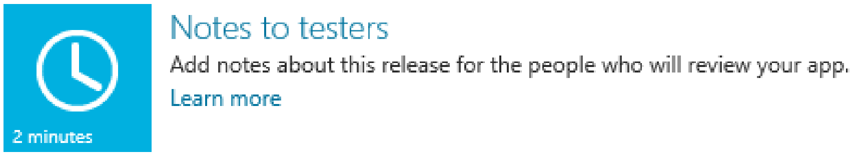
Description (required)	A plain text description of your applications
App features	
Screenshots (at least one required)	Click the Add image link and browse to the screenshots you took earlier (in the *My Pictures/Windows Simulator*) directory. You will also enter some short text for each screen shot.
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, *your-publisher-name*"

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.

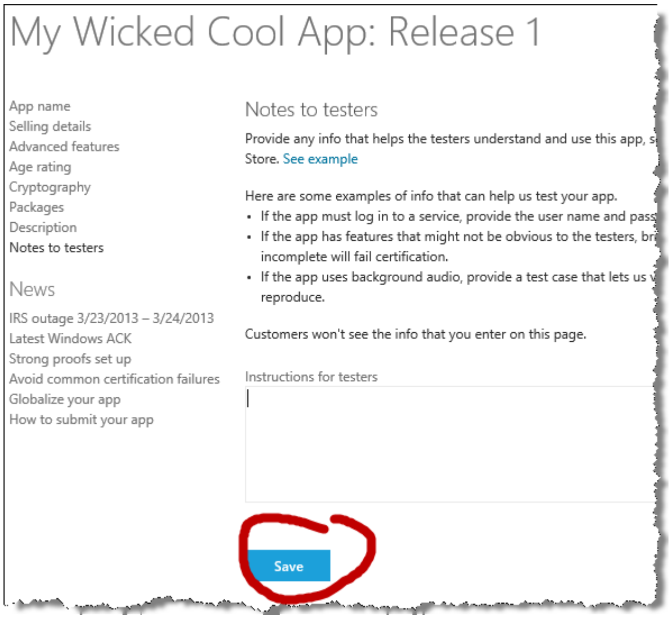
36. Finally, press the Save button:



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



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



My Wicked Cool App: Release 1

Notes to testers

Provide any info that helps the testers understand and use this app, s
Store. See example

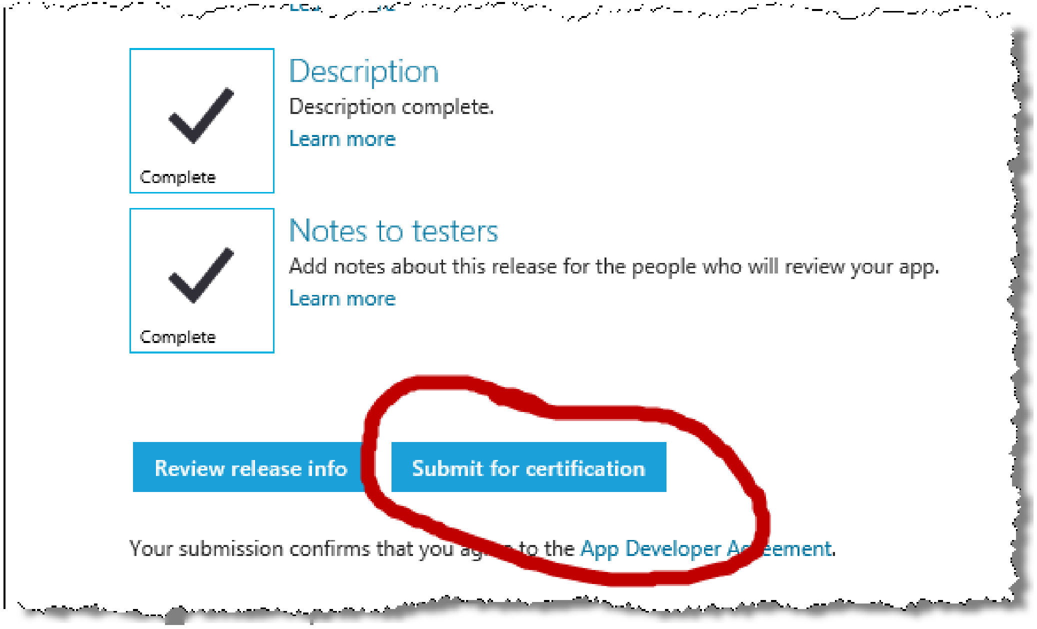
Here are some examples of info that can help us test your app.

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

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

Instructions for testers

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



Description

Description complete.

Learn more

Notes to testers

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

Learn more

Your submission confirms that you ag… to the App Developer Ag…eement.

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!