

Deploying Unity to Windows 8 Store Notes

Wednesday, December 11, 2013 5:34 PM

A "player" to Unity is a build that is built specifically to be deployed to a certain operating system: win7, win8, Linux, etc.

Software needed

To deploy to the Windows 8 store, and if it is not for educational purposes (i.e. for money or a portfolio) you'll need:

- To be running on a Windows 8 or 8.1 (especially if you're doing an 8.1 build). A VM works
- Unity Pro
- Visual Studio 2013

To Build!

In Unity

- File
 - Build Settings
 - Make sure all your scenes for your game are included in the build and in the right order. (Can drag and drop them in and to rearrange the order)
 - Windows Store Apps
 - select win 8.0 or win 8.1
 - ◆ You can build for both or one or the other. From what we saw if you build for win8.1 it isn't backwards compatible.
 - ◆ If you want to have win8 and win8.1, you have to do (from what we saw at least) a build for each and go through all the steps for both of those builds.
 - D3D11 C++
 - Player Settings
 - Publishing Settings
 - ◆ Compilation Overrides
 - ◇ None
 - Set whatever images you can here. There will be more to insert once you open the project in VS2013.

In Visual Studio 2013

For win8 store app build

- **Build configuration:**
 - Change build to "master"

There are three build configurations you can choose from. Debug should obviously be used to debug your scripts. Release optimizes code for better performance. Master configuration build should be used to submit your app to the Store. It has profiler support stripped out.

From <<http://stackoverflow.com/questions/20122522/whats-the-difference-between-a-release-and-a-master-in-vs>>

More info: <https://docs.unity3d.com/Documentation/Manual/wp8-deployment.html>

- **Solution Explorer**
 - Open "Package.appxmanifest"
 - Update names and icons as necessary
 - When creating the package manifest remember not to have any spaces in any of your art or names. Your name will change when it gets to the store. **More info:** <http://social.msdn.microsoft.com/Forums/windowsapps/de-DE/e3cc558b-b1a8-4f99-b180-83f070cd2dc7/validation-error-we-cant-read-the-appxbundle-while-uploading-the-package?forum=windowsstore>
- **In the menu bar go to**
 - Project
 - Store
 - "Create App Package"
 - ◆ Make sure you have your win8 dev account set up as you'll have to login to your account in order to complete the process (i.e. already have your app associated with the store before you upload)
- You can run optional cert tests on your package after you've made the build that will better tell you if you'll pass cert. But not a guarantee either way. You can also skip it
 - If you run them the app you made will load and close multiple times. Don't interact with the app while the test is running. It will mess with the test and make it fail.
- GOOD LUCK!!

For deploying to a tablet:

- **Build configuration:**
 - Use build as debug or release (optimized) for debugging, "master" if a final release
 - **Solution Explorer**
 - Open "Package.appxmanifest"
 - Can change settings if wanted (like the icon, etc.). Not necessary for debugging unless you're testing out the images and changes in name to see what you like.
 - **In the menu bar go to**
 - Project
 - Store
 - Create App Package
 - ◆ If NO (not to store) -- i.e. deploying to tablet
 - ◇ next
 - ◇ x86
 - ▶ Create
 - ▶ Close
- **In file explorer:**
 - Open where app package got stored in file explorer
 - "AppPackages"
 - ignore *.appxupload
 - Zip the folder
 - download zipped folder to the tablet.
 - extract unzipped folder
 - right-click on "Add appdev package"
 - run with powershell
 - ◆ this installs it. Follow the prompts