Intel® HTML5 Development Environment

Tutorial – Building a Google* Android Application Binary

V1.02 : 08.08.2013

Legal Information

INFORMATION IN THIS DOCUMENT IS PROVIDED IN CONNECTION WITH INTEL PRODUCTS. NO LICENSE, EXPRESS OR IMPLIED, BY ESTOPPEL OR OTHERWISE, TO ANY INTELLECTUAL PROPERTY RIGHTS IS GRANTED BY THIS DOCUMENT. EXCEPT AS PROVIDED IN INTEL'S TERMS AND CONDITIONS OF SALE FOR SUCH PRODUCTS, INTEL ASSUMES NO LIABILITY WHATSOEVER AND INTEL DISCLAIMS ANY EXPRESS OR IMPLIED WARRANTY, RELATING TO SALE AND/OR USE OF INTEL PRODUCTS INCLUDING LIABILITY OR WARRANTIES RELATING TO FITNESS FOR A PARTICULAR PURPOSE, MERCHANTABILITY, OR INFRINGEMENT OF ANY PATENT, COPYRIGHT OR OTHER INTELLECTUAL PROPERTY RIGHT.

A "Mission Critical Application" is any application in which failure of the Intel Product could result, directly or indirectly, in personal injury or death. SHOULD YOU PURCHASE OR USE INTEL'S PRODUCTS FOR ANY SUCH MISSION CRITICAL APPLICATION, YOU SHALL INDEMNIFY AND HOLD INTEL AND ITS SUBSIDIARIES, SUBCONTRACTORS AND AFFILIATES, AND THE DIRECTORS, OFFICERS, AND EMPLOYEES OF EACH, HARMLESS AGAINST ALL CLAIMS COSTS, DAMAGES, AND EXPENSES AND REASONABLE ATTORNEYS' FEES ARISING OUT OF, DIRECTLY OR INDIRECTLY, ANY CLAIM OF PRODUCT LIABILITY, PERSONAL INJURY, OR DEATH ARISING IN ANY WAY OUT OF SUCH MISSION CRITICAL APPLICATION, WHETHER OR NOT INTEL OR ITS SUBCONTRACTOR WAS NEGLIGENT IN THE DESIGN, MANUFACTURE, OR WARNING OF THE INTEL PRODUCT OR ANY OF ITS PARTS.

Intel may make changes to specifications and product descriptions at any time, without notice. Designers must not rely on the absence or characteristics of any features or instructions marked "reserved" or "undefined". Intel reserves these for future definition and shall have no responsibility whatsoever for conflicts or incompatibilities arising from future changes to them. The information here is subject to change without notice. Do not finalize a design with this information.

The products described in this document may contain design defects or errors known as errata which may cause the product to deviate from published specifications. Current characterized errata are available on request. Contact your local Intel sales office or your distributor to obtain the latest specifications and before placing your product order. Copies of documents which have an order number and are referenced in this document, or other Intel literature, may be obtained by calling 1-800-548-4725, or go to: http://www.intel.com/design/literature.htm

Intel and the Intel logo are trademarks of Intel Corporation in the U.S. and/or other countries.

*Other names and brands may be claimed as the property of others.

Microsoft, Windows, and the Windows logo are trademarks, or registered trademarks of Microsoft Corporation in the United States and/or other countries.

Copyright © 2013, Intel Corporation. All rights reserved.

Purpose

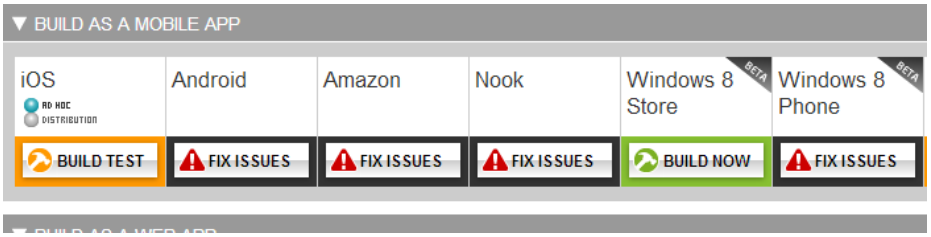
This document will show you, the developer, how to take your application written using the Intel® XDK and generate an Google* Android binary file. This document shows the Google Android build screens, however, the same steps are used for Amazon* and Barnes & Noble Nook* builds.

You can also watch a video tutorial that walks you through the build process for Android builds at the following URL:
http://www.html5dev-software.intel.com/documentation/index.php?DOC=ANDROID_BUILD_VIDEO

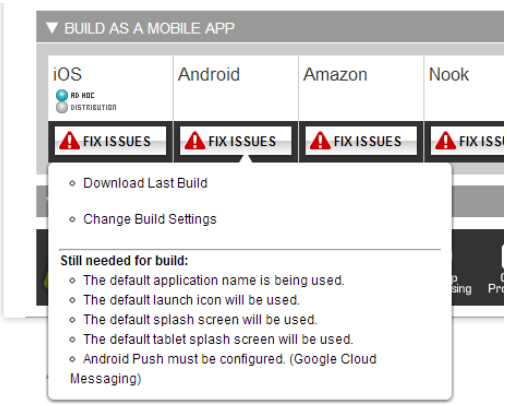
Building a Google* Android Binary

Once you are ready to create an actual .APK file for your Android device, open the Intel XDK to your application and click the *Build for App Store* icon (pictured to the right) that is found across the top of the Intel XDK. If you want to promote your latest QA bundle to production prior to building, select the *Upload & Build* option; otherwise select *Build Only*.

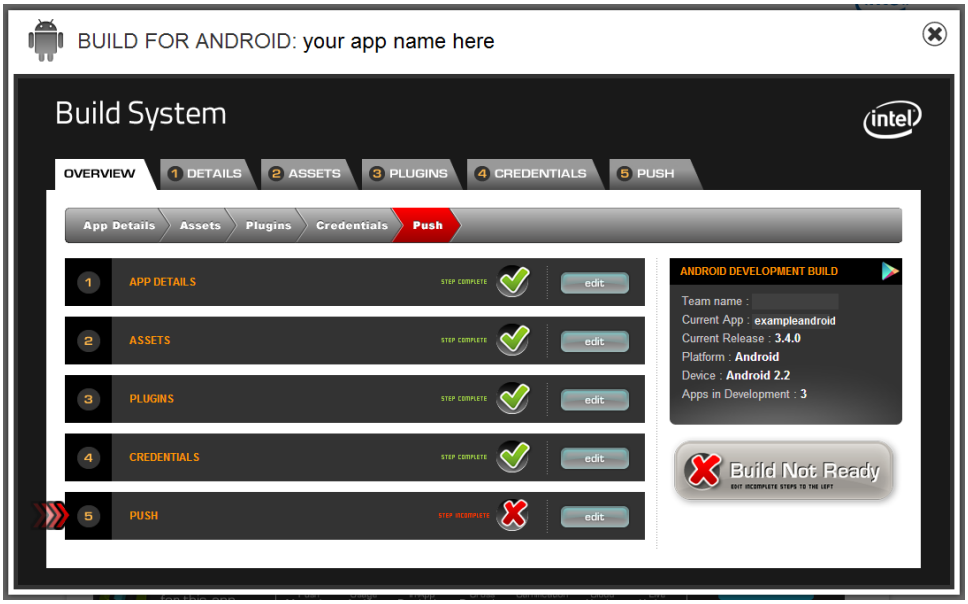
The Intel XDK will take you to the App Dev Center on the Control Center screen with your application selected and the *Build as a Mobile App* section expanded.



Here, you will be able to automatically build if everything has been set up previously (indicated by a green *Build Now* indicator), or enter the Build system to provide necessary information for building. The latter condition will be indicated by an orange *Build Test* or red *Fix Issues* indicator. In either of these modes, select the Change Build Settings link which will take you to the build system *Overview* screen for your application.

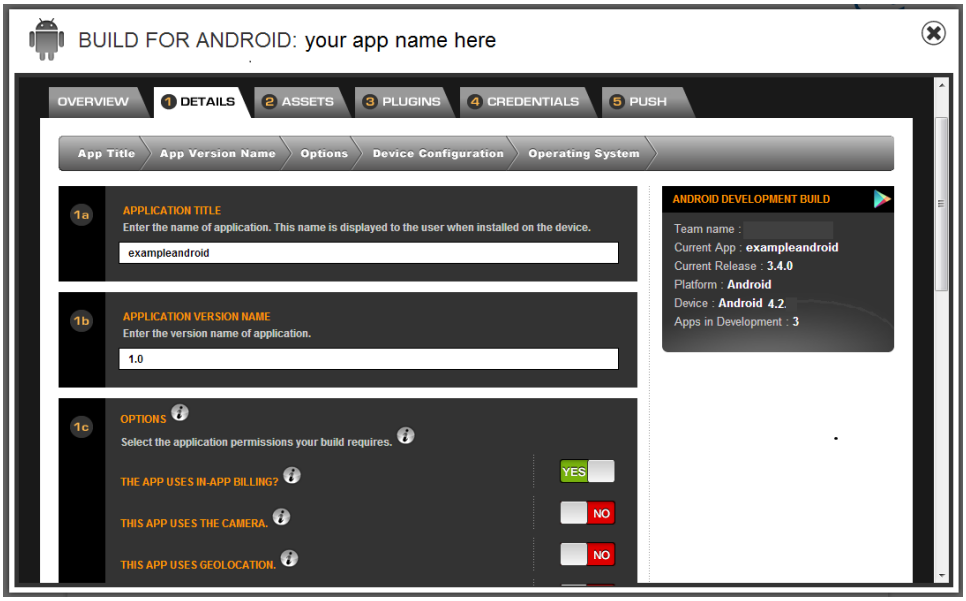


Android Build System



Steps that are already completed for a build will be indicated by a green check, while items that require attention in order to build are indicated by a red X. For this tutorial, we will walk through each step individually.

Details



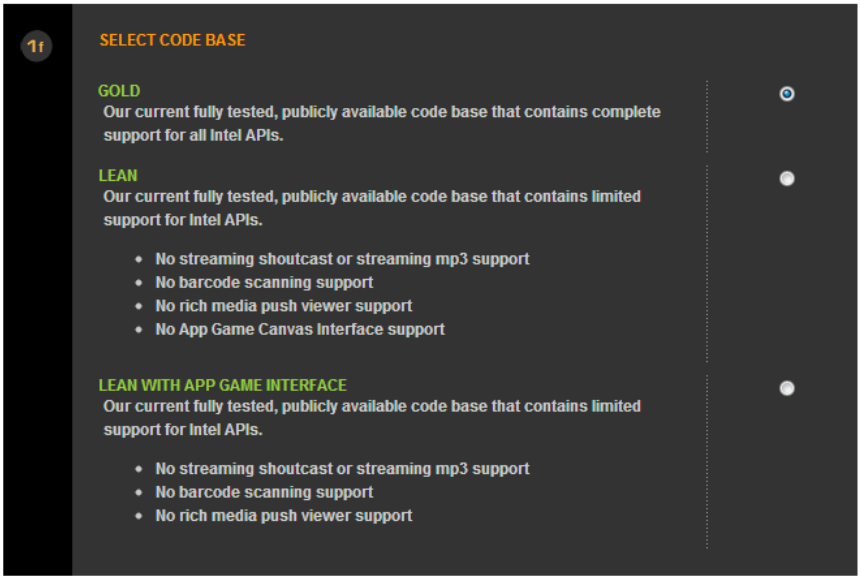
On the Details tab, you will be asked to provide the title of your application as it will appear on the device. By default, the rightmost portion of your project name is used. For example, if your project name is *testuser.exampleandroid*, the default app name will be *exampleandroid*. Be aware that on some devices, if the name is too long it will be truncated with an ellipsis or broken to a second line. As a rule of thumb, we suggest keeping the application name less than 19 characters.

You will also need to provide a version number for this build of the application. You may set this number so that you can track which version of the application is in the market.

You will be asked a list of questions about the application's permissions. Answer the questions appropriately to allow all the functionalities required by your application while removing any permission that is not needed. By default, applications are built with permissions to all functionalities of the device. You can alter these build permissions by turning them on/off on this build screen.

You will also provide the device configuration and operating systems your app is targeting. Currently 4.2 is the only Operating System choice, however your app should run on any Android device using Android 2.3+.

Finally, you will select the code base you wish to build with.



There are three (3) options available:

- **GOLD** – the current fully tested, publicly available code base that contains complete support for all Intel APIs.
- **LEAN** – the current fully tested, publicly available code base that contains limited support for Intel APIs. Items NOT supported are streaming shoutcast or mp3 support, barcode scanning support, rich media push viewer support, and App Game Interfaces support.
- **LEAN WITH APP GAME INTERFACE** – the same as LEAN but App Game Interfaces are supported.

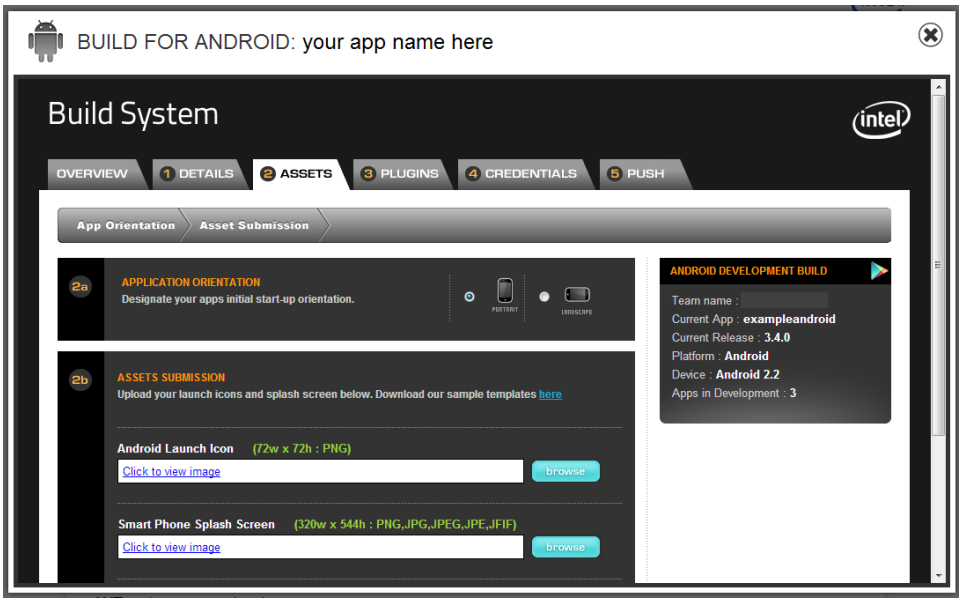
Choosing either of the LEAN options will reduce the size of your binary. One of these is recommended if your application does not require support for the excluded items.

Once you have completed all items, select the

button. This will take you to the Assets tab.

Assets

In order to complete your build, you will need a series of assets such as launch icons and splash screens.



On this tab, you have an option to create your application for landscape or portrait mode. Once you have the initial orientation set, you will need to upload the appropriate assets for the application. By default, a set of Intel XDK icons are used. This allows you to build and test without creating your own specific assets. Once you are ready to publish your application, you can upload your own asset files from this screen. As a starting point, you can download sample templates and directions for creating your own. The size and file format specifications for each asset are provided on the screen.

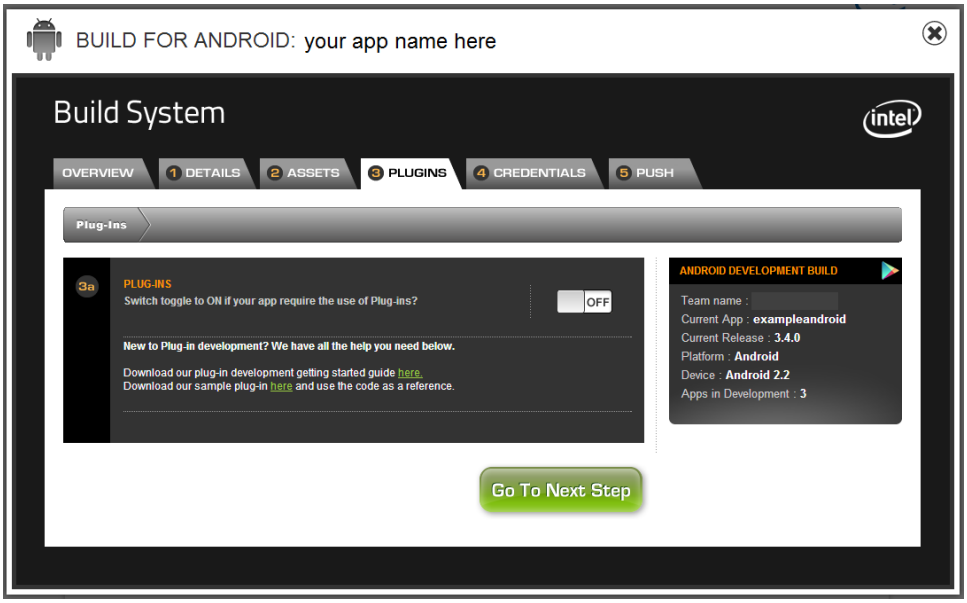
Once you have completed all items on this screen, select the

button. This will take you to the Plugins tab.

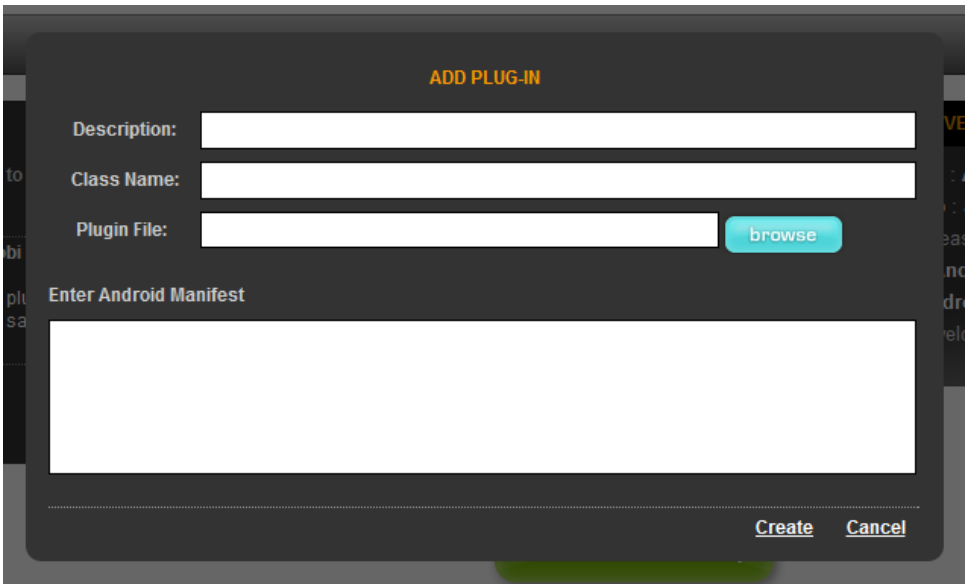
Plugins

This section is used to include custom native code known as "plugins" into your build. To learn more about plugins, download the plugin documentation here:

http://www.html5dev-software.intel.com/documentation/index.php?DOC=PLUGIN_DOCUMENT.



If your application requires the use of plugins, set the toggle switch to ON. Next, click the add another plugin button which takes you to a screen where you can add information about your plugin and create it.

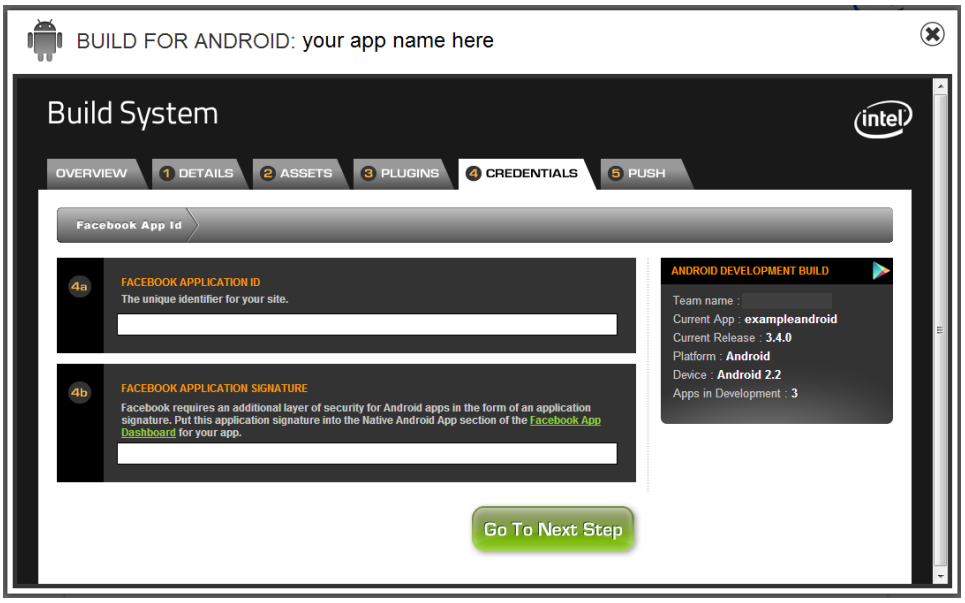


After either specifying the plugins you wish to include, or opting to add none, select the Go To Next Step button. This will take you to the Credentials tab.

Credentials

On this tab, you will provide your Facebook credentials if you want your app to be available as a Facebook application. You will provide your Facebook Application ID which is required, and Facebook

requires an additional layer of security for Android apps in the form of an application signature. A signature is provided that you may then use in the Native Android App section of the Facebook App Dashboard.

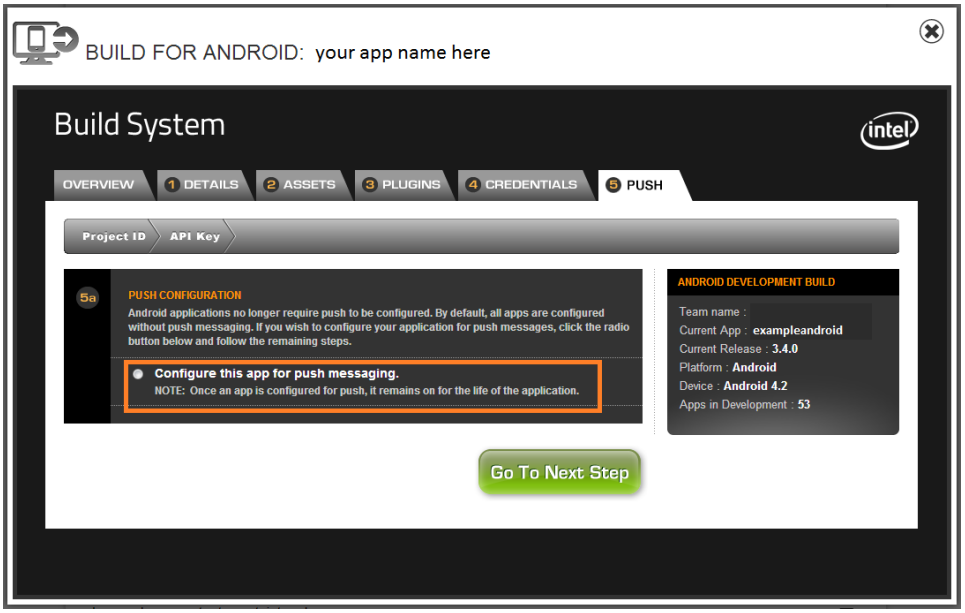


Once you have completed all items on this screen, select the

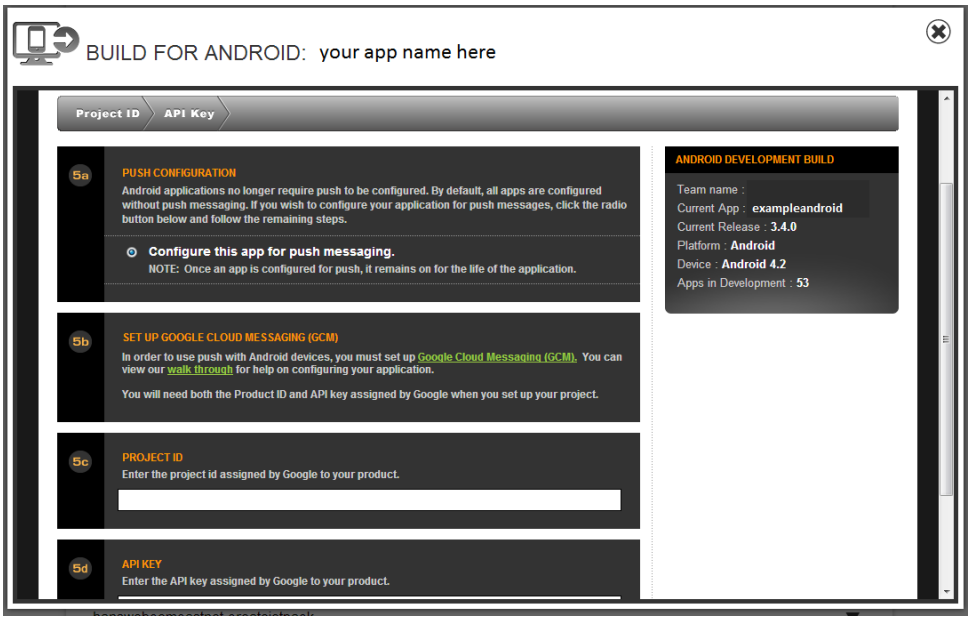
button. This will take you to the Push tab.

Push

By default, all new apps are configured without push messaging. In order to use push messaging with Android devices, you will have to set up Google Cloud Messaging* (GCM). To do so, click the radio button for *Configure this app for push messaging*. Otherwise, proceed to the next step.

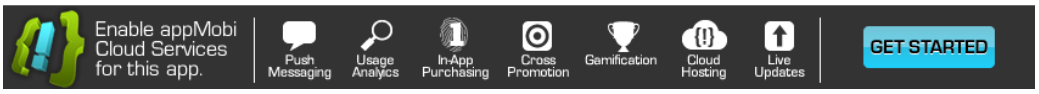


If you detemine you want to configure push messaging, the screen will be expanded to allow you to enter additional information required. We have provided a couple of links in step 5b to help you with your GCM set up. A link to the GCM web page, and a “walkthrough” link which takes you through the process.



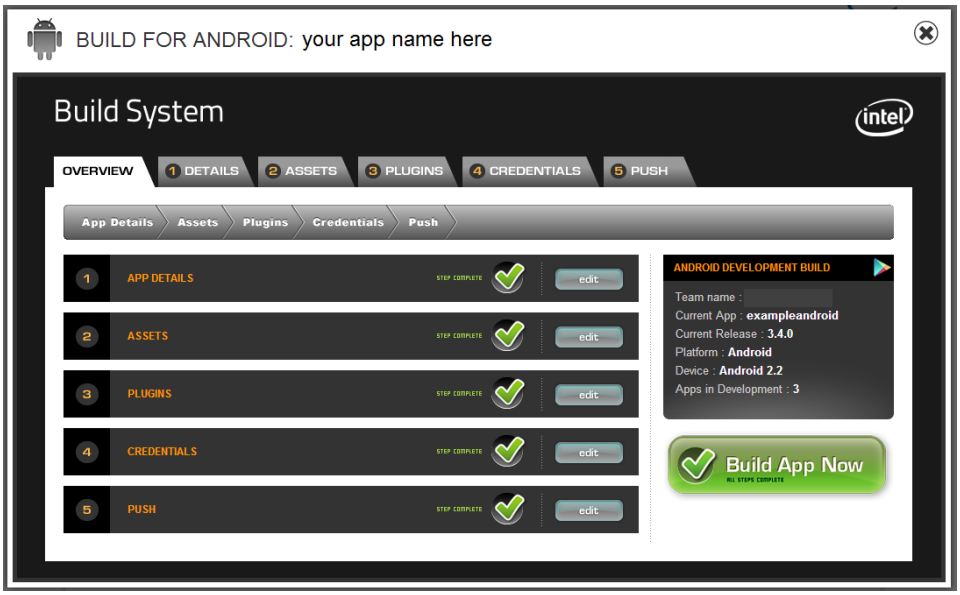
When you set up GCM for your app, Google will provide you with a Project ID and API Key. Enter those in the appropriate fields on the Push screen.

> Note: You will also have to enable appMobi cloud services from the App Dev Center by clicking the Get Started button associated with the application you are building.



Once you have completed all items on this screen, congratulations, you are ready to build. Select the

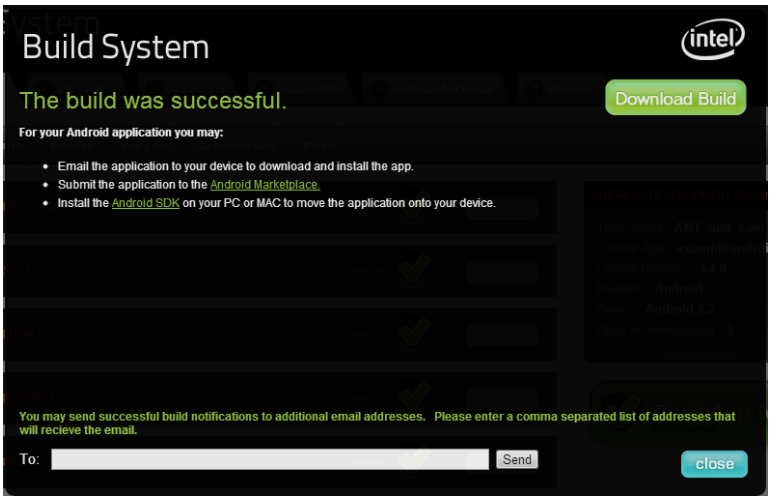
button. This will take you back to the to the Overview tab, and the Build App Now button should be available.



Build Your App

Select the Build App Now button from this screen. A status indicator will show that the build is taking place.

Once your binary is successfully built, a Build Successful screen is shown.



From here you can download your application by clicking on the Download Build button, transfer the .APK file to your Android device by copying it over its wired connection or emailing it to your device. Once you have the file on the device, select it to install the test application.

Related Documents

Here are some links to other tutorials describing how to perform builds on other platforms.

Apple* iOS:
http://html5dev-software.intel.com/documentation/index.php?DOC=TUTORIAL_BUILD_BINARY_IOS

Microsoft Windows* Store:
http://html5dev-software.intel.com/documentation/index.php?DOC=TUTORIAL_BUILD_PACKAGE_WINDOWS_STORE

Microsoft Windows Phone 8*:
http://html5dev-software.intel.com/documentation/index.php?DOC=TUTORIAL_BUILD_PACKAGE_WINDOWS_PHONE_8