

Introduction

Mag+ is a platform that allows you to create Android tablet-optimized publications. The platform includes a plugin for Adobe InDesign, where you design your publication; a production tool, where you assemble it; and a Web-based admin tool where you build your branded app and upload and prepare your issues for distribution through the app. Google Android Market or another Android market provides the distribution of the app and all e-commerce in the Android Market and in-app purchases.

To publish a mag+ production on the Android Market the following steps are required:

- Create an Android Keystore for your signing certificate
- Create the Application .apk file for the publication in Mag+ Publish
- Test install the Application .apk file on the tablet
- Submitting the publication to Android Market
- For Android Market in-app purchases, two more steps are required:
- Configuring the Android Market public signing key in Mag+ Publish
- Configuring products and prices for Android Market single issue in-app-purchases

Remarks about Android Market

When submitting a publication to the Android Market, you can never change an application's publishing pricing option from Free to Paid so make sure that you select the Paid option for non free content.

When submitting the publication there is a number of graphical assets that are either required or optional, these assets are used for promotion so best practice is to make sure that you have prepared and created all assets before you submit. Follow this link (requires Android Market login) for a more detailed description of marketing material [Android Market - Graphic Asset for your Application](#)

Limitations to the Android app compared to the iPad app

- Dual orientation – the current version of the Android app only supports landscape publications. Please make sure that your issues support landscape, they may also support portrait, but may not be locked to portrait only.
- Menu customization – it is not possible to customize the menu to show custom pictures in the menu.
- Language customization – the Android app only support english for the different messages that appears in the app.
- 3rd party services for Tracking, Analytics and Ads – none of the 3rd party services that are available for the Mag+ iPad app are available in the Android ver-

sion (AdMob, Flurry, Localytics, Omniture, TapJoy).

- 3.0 Mag+ features – Some of the new features introduced in the iPad version in release 3.0 are not yet available in the Android version, such as: Sharing, Playlists, Banner view in store, No A-layer and issue update indication
- In app purchases of subscription are not available.

Limitations of the in-app purchase solution

At the moment, for Android Market in-app purchases of issues, we only support the purchase “transaction”. There are two states available from Google, “cancel” and “redeem transaction”, these transactions can be triggered by the user after a transaction has been processed. In this case the customer will get his/hers money back but will keep the issue. However, in the case the user does a “restore” in the application the cancelled and redeemed issues will not be available any more. (restore can also happen in case of a re-installation or clearing of the application)

For Android Market in-app purchases we are not able to display price information in the overview store list as Android Market do not support this functionality. However, once the user goes to the Android Market checkout flow he/she will be presented with the price information before pressing the buy button. To have the price displayed in the store we recommend entering the price in the issue title or the description.

Create an Android Keystore for your signing certificate

First off, you will have to create your private key certificate (also known as “Keystore”):

How to create your Keystore file on a Mac:

Open Terminal

Enter: `keytool -genkey -v -keystore my-release-key.keystore -alias alias_name -keyalg RSA -keysize 2048 -validity 10000`

Below I have pasted a screenshot from Terminal when setting up the Keystore:

```
Support-test-Anders:~ anders$ keytool -genkey -v -keystore my-release-key.keystore -alias NAMEOFKEY -keyalg RSA -keysize 2048 -validity 10000
Enter keystore password:
What is your first and last name?
  [Unknown]:  First Last
What is the name of your organizational unit?
  [Unknown]:  Department
What is the name of your organization?
  [Unknown]:  Organization
What is the name of your City or Locality?
  [Unknown]:  City
What is the name of your State or Province?
  [Unknown]:  State
What is the two-letter country code for this unit?
  [Unknown]:  US
Is CN=First Last, OU=Department, O=Organization, L=City, ST=State, C=US correct?
[no]:  YES

Generating 2,048 bit RSA key pair and self-signed certificate (SHA1withRSA) with a validity of 10,000 days
for: CN=First Last, OU=Department, O=Organization, L=City, ST=State, C=US
Enter key password for <NAMEOFKEY>
(RETURN if same as keystore password):
[Storing my-release-key.keystore]
```

- When you create the Keystore file in terminal you will be asked to create a Key Store password, a Key alias and a Key alias password. These credentials are important to have when setting up the app later.
- If you dont have access to a Mac, follow this guide to set it up on a PC (Read “Signing in Release Mode”): <http://developer.android.com/guide/publishing/app-signing.html#cert>

Create the Application .apk file for the Publication

Before You Begin

- Make sure that you have created a private key certificate.
See: certificate, <http://developer.android.com/guide/publishing/app-signing.html>
- Make sure that you have signed up as an Android Market developer so that you become eligible to submit your apps to the Android Market.
See: Android Market developer, <http://market.android.com/publish/Home>

Walkthrough

Contact sales@magplus.com to gain access to publish.magplus.com.
Go into Publish (publish.magplus.com) and select your publication.

1. Click APPS > Android



Apps

Here you can upload assets to build new applications.

[iPad](#) [Android](#) [Live Settings](#)

Android

Assets and Configuration

[Upload help Issue MIB file](#) **Incomplete**

[Android Settings](#) **Incomplete**

2. Upload HELP MIB file - This is a navigation help that will be embedded with the app. Do not change the name of the MIB created in production tool, this is because the file name must be the same as its specified MIB guid.

Upload Help Issue MIB File

com.magplus.help.1.mib	312.21 kb
------------------------	-----------

[Browse](#)

[Upload](#)

3. Configure the specific “Android Settings”

[Back to Apps](#)

Android Settings

Android Settings

Key store password
 The password used to access the keystore file

Key alias
 The alias name for the key certificate stored in the keystore file

Key alias password
 The alias password used to access the key certificate stored in the keystore file

Package name
 The identifier used to identify the help MIB

Version
 The user-friendly version name of the application using the naming convention x.y.z, for example 1.0.0

Keystore file
 keystore.file The keystore file containing the certificate to be used to sign the application

Icon image file
(png, dimensions: 72x72, required)
 icon.png The icon for the application

4. Press save to store the configuration

5. Create the application signed apk file by pressing “Generate New Build”

Apps

Here you can upload assets to build new applications.

[iPad](#) [Android](#) [Live Settings](#)

Android

Assets and Configuration

Upload help issue MIB file	Complete
Android Settings	Complete

6. Go back/refresh the screen after a while as it takes some time go build the .apk application
7. When the build is finished the apk file will be listed under “Builds” in descending order based on creation time.
In our example above the name of the apk. is:
android-generic-tester_1.0.0-168-release.apk
here the file name consists of:
<brand name>_<user friendly version text>-<unique build identifier>-release.apk
8. Download the newly built application to your desktop by pressing on the link for the apk file.

Continue reading for instructions on how to install the app on your device for testing and then submit the application to Android Market.

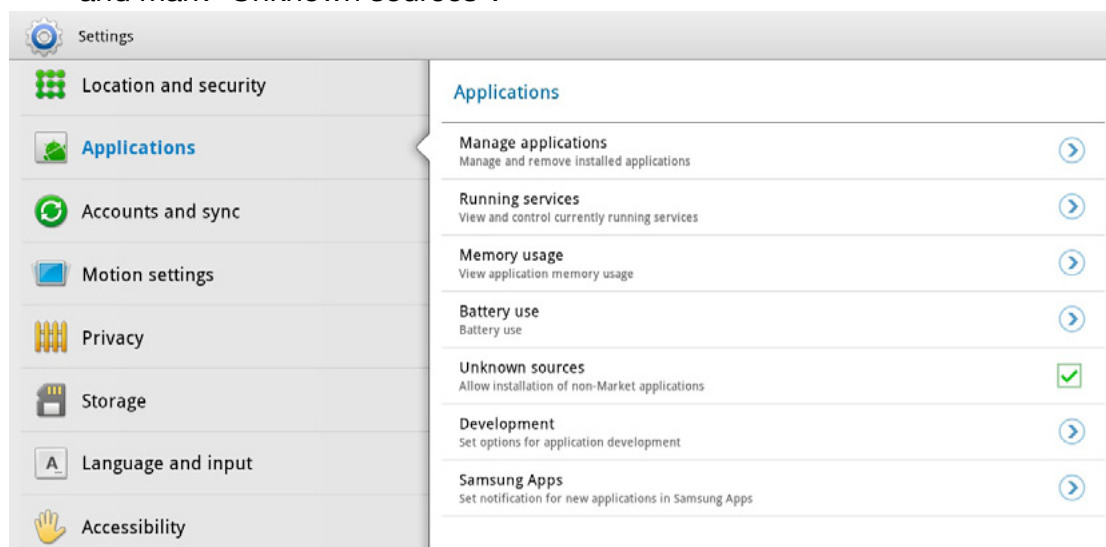
Install the Application .apk file on a tablet

Before You Begin

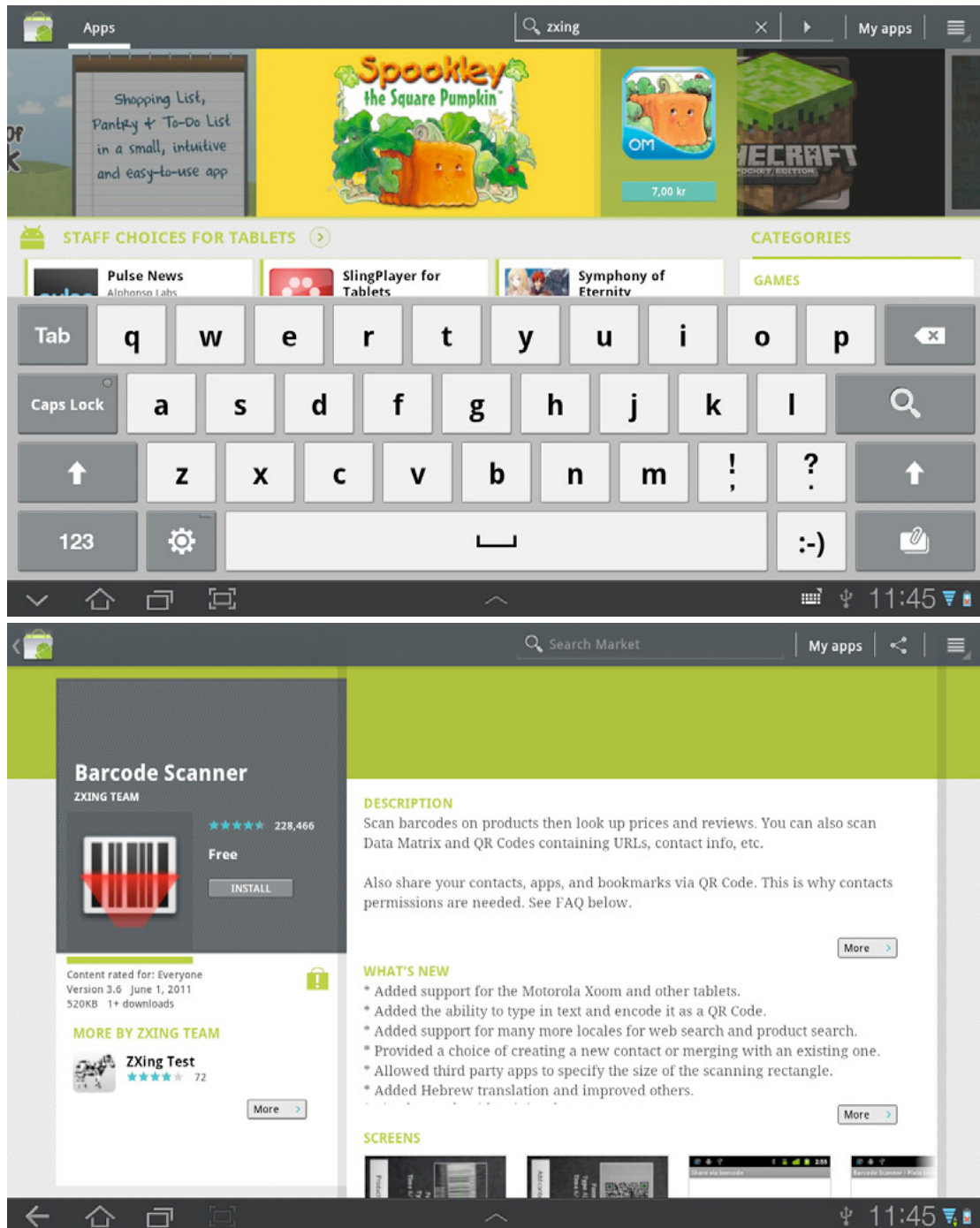
- Make sure that you have setup your Google account on the tablet
- Make sure that you have internet connectivity via your browser on the tablet
- Make sure that you have generated your application .apk file in the Mag+ publish website and stored it on your computer desktop

Steps

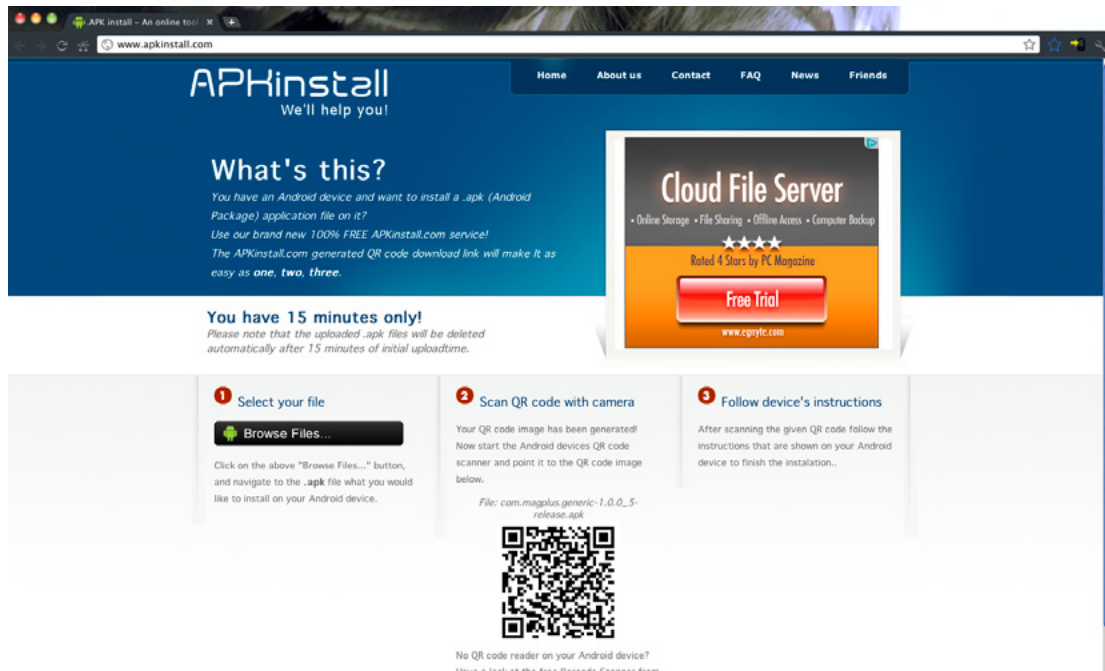
1. Configure the tablet to allow installation of application .apk files outside the Google Android Market - Start the “settings” applications from applications and mark “Unknown sources”.



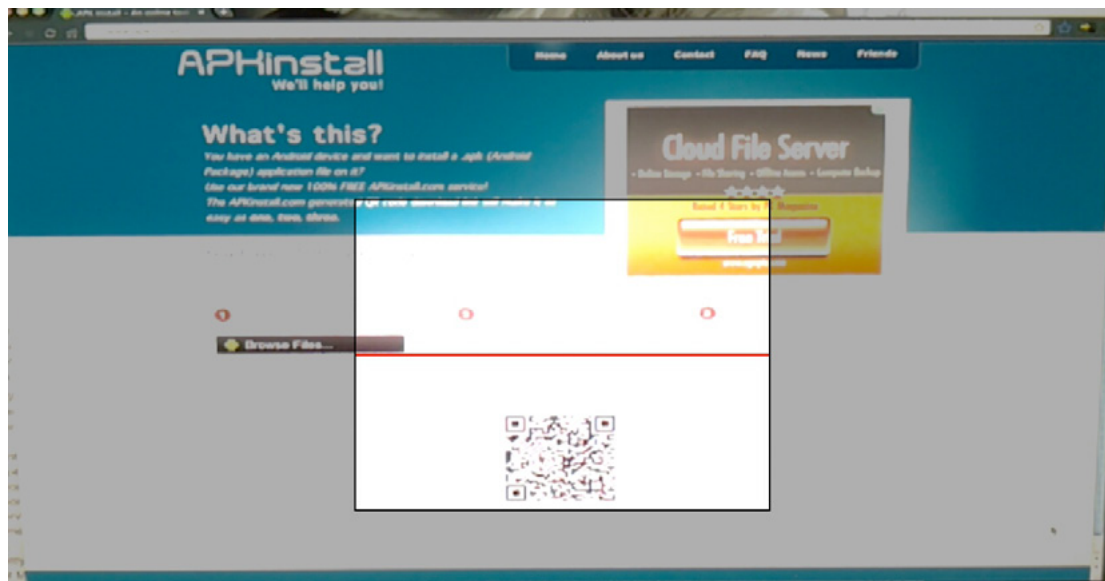
2. Install the ZXing barcode reader from the Google Android Market - Search for ZXing in the Google Android Market application and install the ZXing barcode reader application.



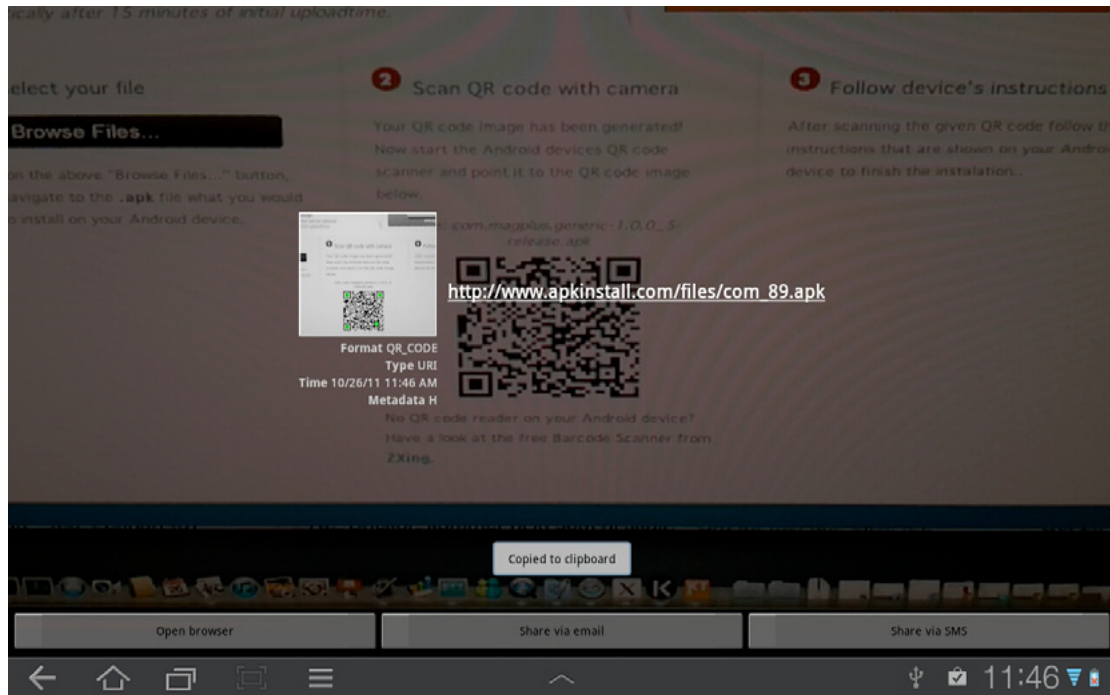
3. On your computer: Go to www.apkinstall.com and upload the Mag+ application APK. Just follow the steps given on the website to upload your apk file. After upload a barcode will appear on the screen. Using the barcode reader we just installed on the tablet we will install the application.



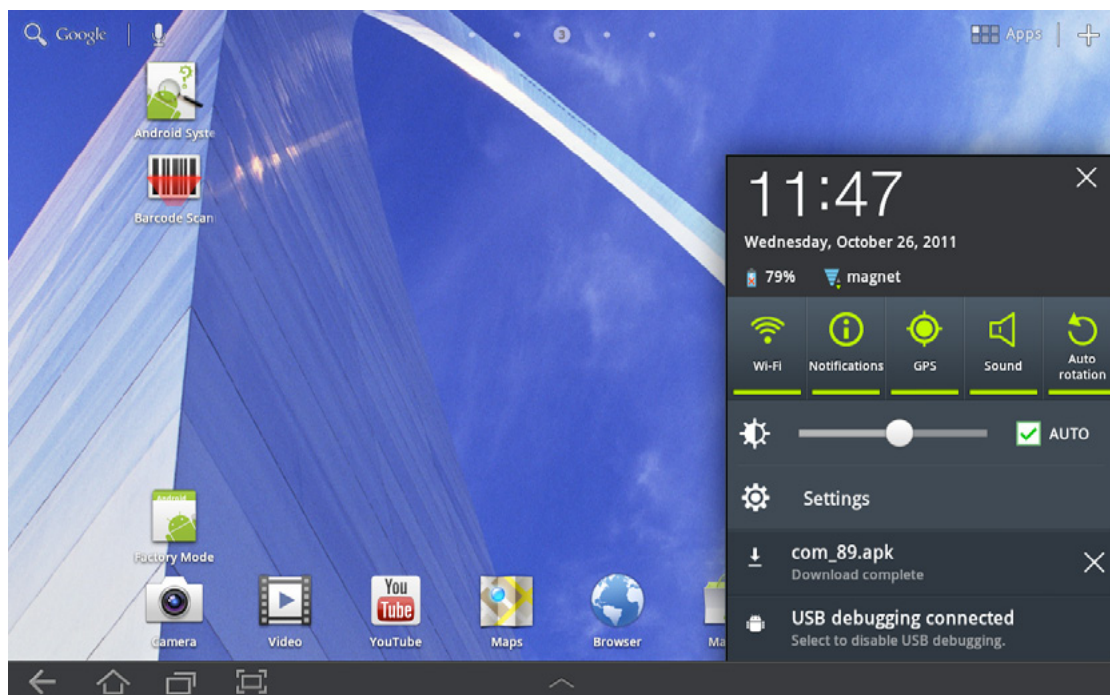
4. Launch the ZXing barcode reader on the tablet. Center the tablet camera on your computer screen over the barcode and wait until it detects the barcode on the computer screen. This can be quite tricky, but if you make sure to point the camera to avoid reflection etc it will eventually detect the barcode.



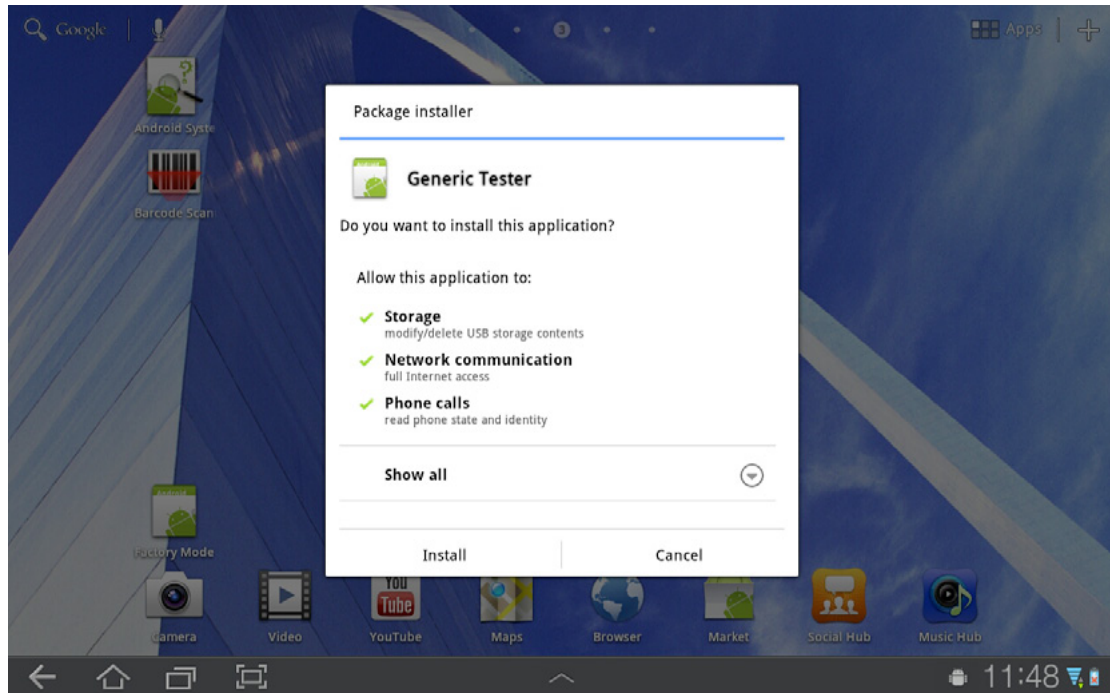
5. Once the barcode is detected press the link in the middle of the tablet window.



6. When the .apk is downloaded you can find it by clicking in the task bar in the bottom right of the tablet.



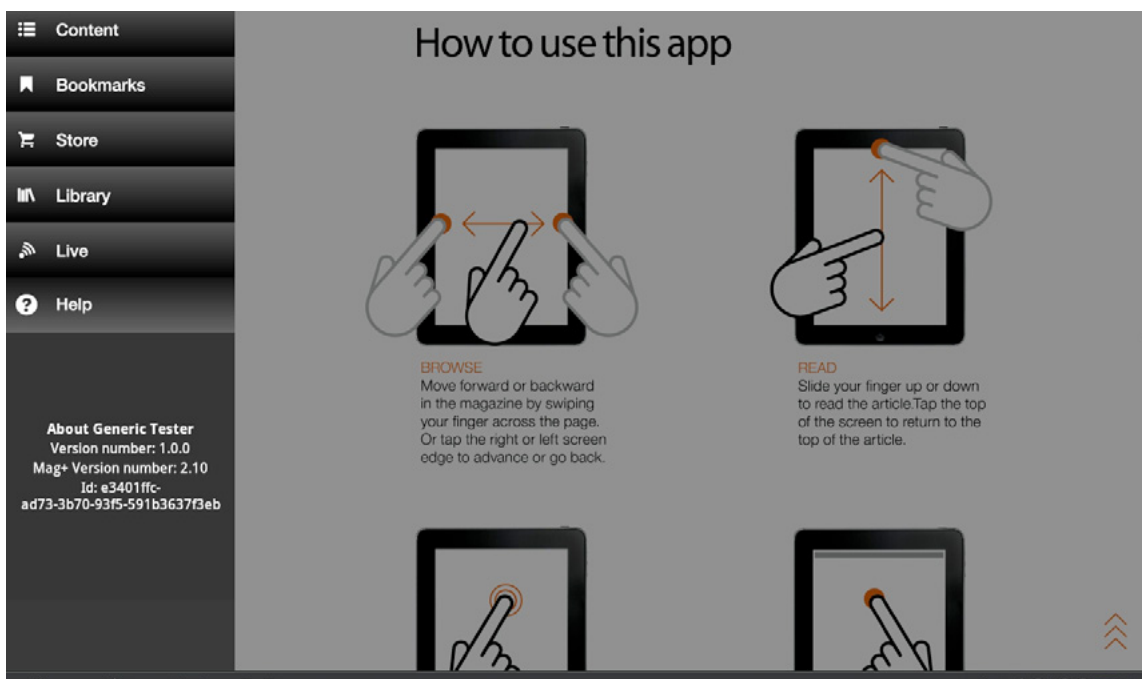
7. Click on your downloaded file, in this case “com_89.apk” and follow the installation instructions.



How to view unpublished issues on the tablet

To preview unpublished issues you need to register your tablet as a test device in the Mag+ publish web site.

1. Locate the test device identifier on the tablet - Launch Mag+ application on the tablet and press the “Help” button to the left.
2. Locate the test device identifier called “Id:” with the format “xxx-yyy-zzz” and write down the sequence on paper.



3. Go to publish.magplus.com, select your publication and register the device with help of the device id. Press Create. Done!



Devices

i Test devices are allowed to read all issues even those that are not published.

New Test Device

Device Identifier (UDID)

To get the identifier for the device, connect the device to iTunes, click the serial number on the summary page until you see the UDID, then select Edit->Copy from the menu.

Owner

Name and/or email

Create

Submitting the Publication to Android Market

Before You Begin

Make sure that you have signed up as an Android Market developer so that you become eligible to submit your apps to the Android Market.

See: Android Market developer, <http://market.android.com/publish/Home>

Make sure that you have done tax setup correctly on your Android Merchant Account:

- No tax
- Fixed tax within EU
- Tax specified per country within EU
- Tax in and outside EU

See: *Selling Your Apps*, <http://www.google.com/support/androidmarket/developer/bin/topic.py?&topic=15867>

Walkthrough

Navigate to <https://market.android.com/publish> and log in.

Click *Upload Application*. You will now be asked to upload your APK file, browse for your apk file you have downloaded from Publish and click “Upload”.

Edit Application - Enter all information asked for.

Note!

For each language you will need to enter the Title of the publication (max 30 characters), enter a description of the publication (max 500 characters) and also enter Promo text for the publication (max 80 characters).

Select *Applications* as Application Type.

Select Off as Copy Protection in Publishing Options, this since the option is “deprecated” when using Google's server. When set to On, at the writing moment, the App does not appear on some tablets.

Select Everyone for Content Rating.

Note you can never change from Free to Paid in pricing!

The screen layouts are listed here http://developer.android.com/guide/practices/screens_support.html. They are changed in the manifest.xml but are basically:

small 2-3 inches	normal 2-5 inches
large 4-7 inches	xlarge 7-12 inches

Create an Android in-app products

If you are planning to sell issues in your app you will have to create in-app products for every issue you want to sell. During this process you will set the price of your issue and also create a Product ID for the product. This Product ID is what you enter in pub-

lish to connect the specific issue with the in-app purchase. If you do not plan to sell issues within your app, ignore this step.

1. Go to <https://market.android.com/publish/Home> and click "In app products" next to your app.
2. Now you will be taken to the in-app Product list for this app, once you have created an in-app product for your app, it will be listed and editable here. Click "Add in-app product" to go to the next step.
3. Now you will be asked to edit the information for your in-app product. You can change or add languages. This is also where you set the price of the in-app product. In our example above we have set the price to 7 SEK and then clicked "Auto Fill" and the other currencies will automatically change. There is a set lowest price for each currency and in some cases you might have to go in and change the price manually in order to over the lowest price in that particular currency.
4. When you are ready here you can choose between to Publish or Save your In-app product. If the app you are creating the in-app product for is unpublished the in-app product will be listed as unpublished until your app is published.
5. When you have saved or published your in-app product you will be taken back your In-app Product list for the application. Here you can go in and edit your in-app product or create new one.
6. To connect the in-app product with an issue go to publish.magplus.com and log in. Select your publication. Click the issues tab. If you have not created an issue yet, click Add new issue. Fill in all information needed, including the product id for the android in-app product you jsut created. If you have already created an issue. Click edit next to it to edit your prodcut id. Upload a MIB and publish!