Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

PassWallet Developer Guide
v0.7

Registered Company Number: 06428260

Registered VAT Number: GB 930 4201 71

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Contents

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Revision History

0.1	Andy Nugent	12th Dec 2012	Initial draft.
0.2	Andy Nugent	13th Dec 2012	Added in example cUrl commands. Added in bulk pass updates.
0.3	Andy Nugent	9th Apr 2013	Added in link to download from Google Play.
0.4	Andy Nugent	28th August 2013	Added in section detailing how to launch PassWallet from an Android app.
0.5	Andy Nugent	28th August 2013	Added section on branding of links for passes.
0.6	Andy Nugent	30th August 2013	Added URL naming section and corrected Android code.
0.7	Andy Nugent	19th March 2014	Added in section for linking passes to Android apps.

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Push Notifications

Registering a Device to Receive Push Notifications for a Pass

In addition to the web service detailed in the Apple document "Passbook Web Service Reference", we require an additional endpoint for registering PassWallet devices.

This has been kept similar to the standard Apple registration endpoint in order to reduce the work required, but is created as a new endpoint so as to not cause potential issues with services that do not support Attido's update method.

Registration is performed via a POST request to:

webServiceURL/*version*/devices/*deviceLibraryIdentifier*/registrations_attido/*passTypeIdentifier*/*serialNumber*

Parameters

- **webServiceURL** - The URL to your web service, as specified in the pass.
- **version** - The protocol version. Currently, v1.
- **deviceLibraryIdentifier** - A unique identifier that is used to identify and authenticate this device in future requests.
- **passTypeIdentifier** - The pass's type, as specified in the pass.
- **serialNumber** - The pass's serial number, as specified in the pass.

Header

The Authorization header is supplied; its value is the word "AttidoPass", followed by a space, followed by the pass's authorization token as specified in the pass.

Payload

The POST payload is a JSON dictionary, containing two key and value pairs:

- **pushToken** - The pushtoken that the server can use to send push notifications to this device.
- **pushServiceUrl** - The URL of Attido's web service for posting update notifications to devices.

Response

- If the serial number is already registered for this device, return HTTP status 200.
- If registration succeeds, return HTTP status 201.
- If the request is not authorized, return HTTP status 401.
- Otherwise, return the appropriate standard HTTP status.

Discussion

The URL of Attido's server should be stored on a per registration basis, and used when an update to the pass occurs.

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Pushing a pass update notification to a device

In order to notify the device that an update for a pass is available the pass provider should POST a request to:

pushServiceURL/*version*/pushUpdate

Parameters

- **pushServiceUrl** - The URL of Attido's web service, as specified in the registration.

Payload

The POST payload is a JSON dictionary, containing two key and value pairs:

- **passTypeID** - The type of the pass that has an update available.
- **pushToken** - The pushtoken that the server can use to send push notifications to this device.

Response

- If the push notification succeeds, returns HTTP status 200.
- If the push notification fails returns 400, and the response body contains details of the error.

Example

To post the following JSON:

```
{
   "passTypeID":"pass.attidomobile.test",
   "pushToken":"cbc8f59160b949199b40ffe21cd2253ef893e4b244944b76ba31b8eaa5f09e69"
}
```

We'd use this cUrl command:

```
curl --data
"{\"passTypeID\":\"pass.attidomobile.test\",\"pushToken\":\"cbc8f59160b949199b40ffe21cd2253ef8
93e4b244944b76ba31b8eaa5f09e69\"}" http://proxy.ravensoft.co.uk/PassWallet/v1/pushUpdate
```

And get this response:

```
{"status":"OK"}
```

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Pushing a pass update notification to multiple devices

In order to notify multiple devices that an update for a pass is available the pass provider should POST a request to:

pushServiceURL/*version*/pushUpdates

Parameters

- **pushServiceUrl** - The URL of Attido's web service, as specified in the registration.
- **version** - The protocol version. Currently, v1.

Payload

The POST payload is a JSON dictionary, containing two key and value pairs:

- **passTypeID** - The type of the pass that has an update available.
- **pushTokens** – An array of pushtokens that the server can use to send push notifications to the devices.

Response

- If the push notification succeeds, returns HTTP status 200.
- If the push notification fails returns 400, and the response body contains details of the error.

Example

To post the following JSON:

```
{
   "passTypeID":"pass.attidomobile.test",
   "pushTokens":["cbc8f59160b949199b40ffe21cd2253ef893e4b244944b76ba31b8eaa5f09e69",
        "a2ee9981afd84d77aa1a2bcf93df49c27d5df254c139458bdace96f7fd50088a",
        "b8129fb4ebf5423b985075b62f9322a8d67bdec69d3c441bd871f12ede46c93c"
        ]
}
```

We'd use this cUrl command:

```
curl --data
"{\"passTypeID\":\"pass.attidomobile.test\",\"pushTokens\":[\"cbc8f59160b949199b40ffe21cd2253e
f893e4b244944b76ba31b8eaa5f09e69\",\"a2ee9981afd84d77aa1a2bcf93df49c27d5df254c139458bdace96f7f
d50088a\",\"b8129fb4ebf5423b985075b62f9322a8d67bdec69d3c441bd871f12ede46c93c\"]}"
http://proxy.ravensoft.co.uk/PassWallet/v1/pushUpdates
```

And get this response:

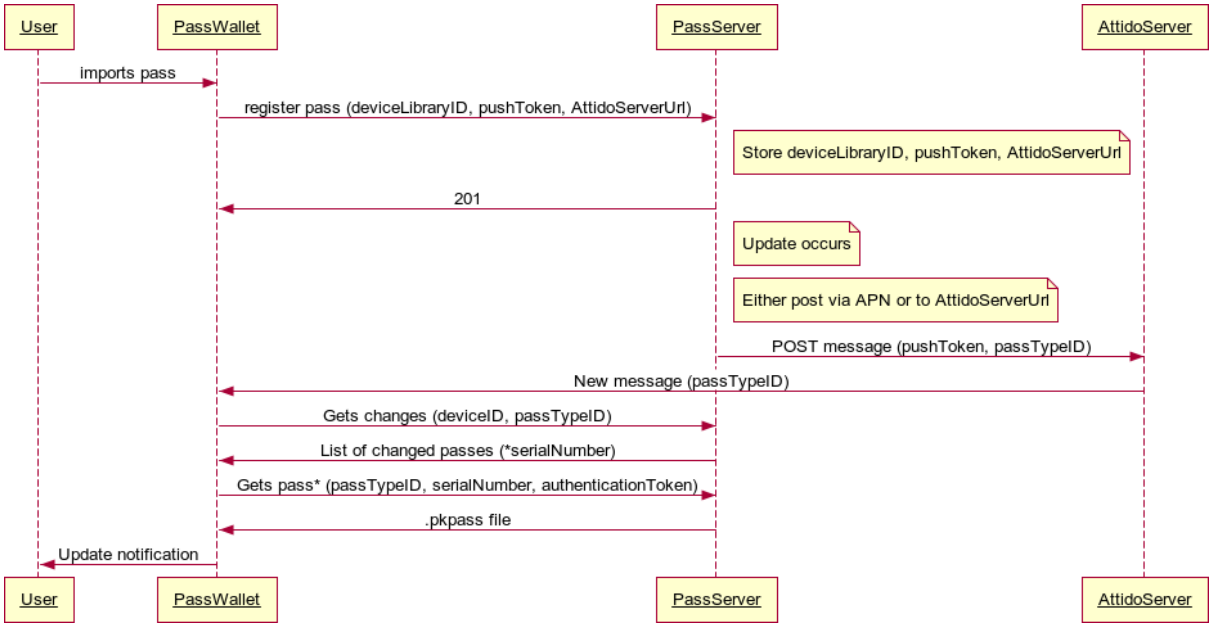
```
{"status":"OK"}
```

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Sequence

The sequence for updating a pass is as follows:



As you can see, once the notification has been passed to the Attido server and subsequently delivered to the device, the following calls are indistinguishable from the sequence seen when an update occurs from an iOS6 based device.

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

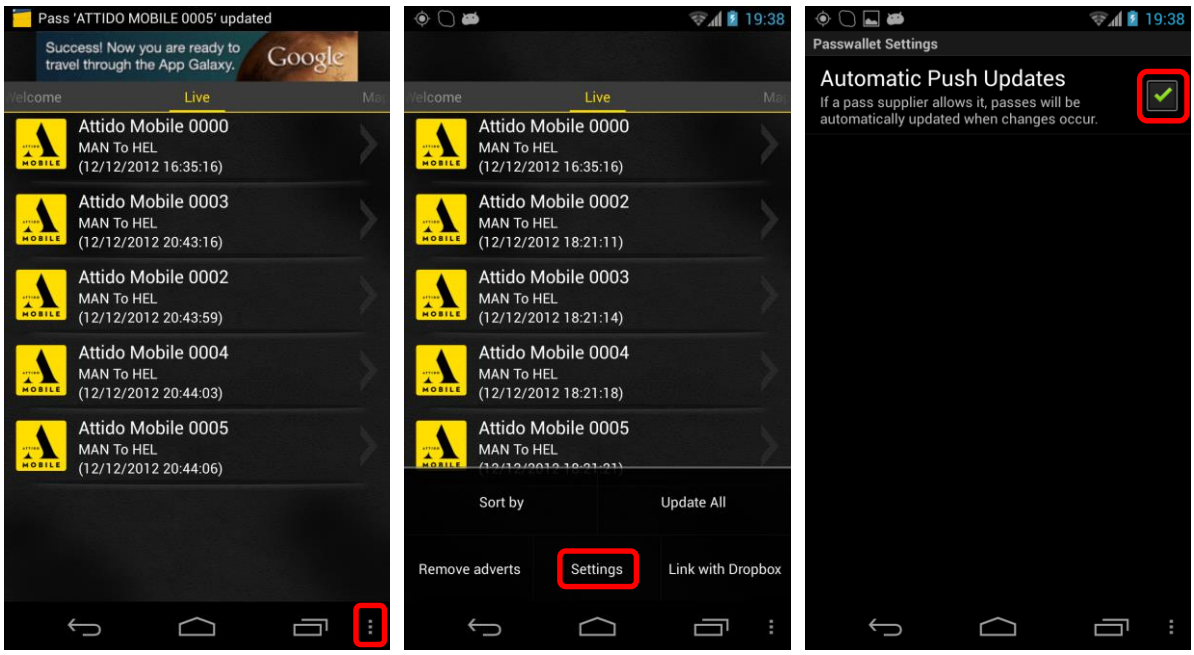
Tel: 0161 4089342 Web: www.attidomobile.com

Testing

In order to test the push notifications, install PassWallet from Google Play:

https://play.google.com/store/apps/details?id=com.attidomobile.passwallet

Make sure that push updates are enabled system wide (default is enabled):



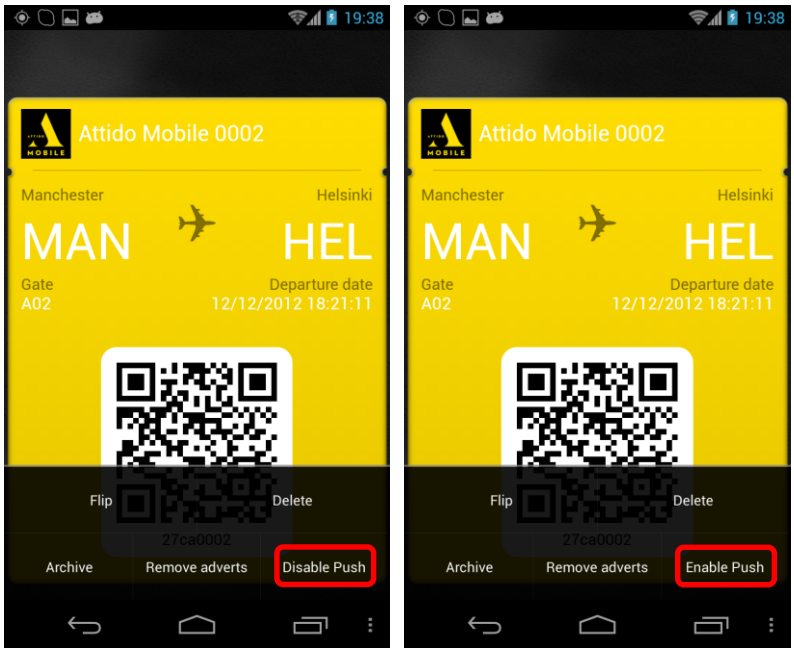
Registered Company Number: 06428260

Registered VAT Number: GB 930 4201 71

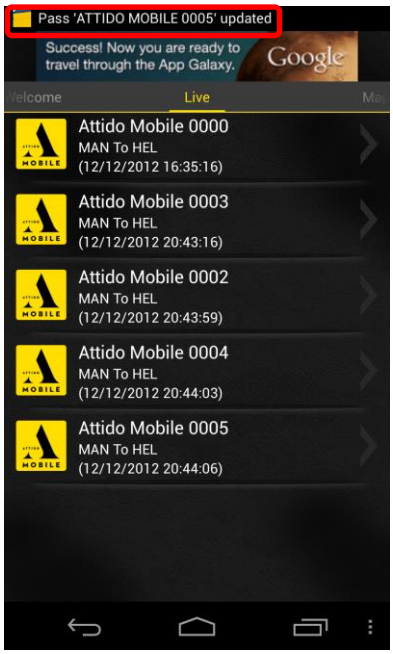
Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

You then have an option to enable or disable push updates on a per pass basis (if the command 'disable push' is shown, then push is currently enabled and vice versa):



When a push notification arrives at the device, notifications are displayed to the user as passes are updated:



ATTIDO MOBILE

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Launching PassWallet from an Android app

The following code will launch PassWallet from an Android app if installed, or launch Google Play if not.
The "referrer" will be the URL of the pass you're trying to install and will be automatically imported on the device when PassWallet is first run.

```
private static boolean launchPassWallet(Context applicationContext, Uri uri, boolean launchGooglePlay) {

    if (null != applicationContext) {

        PackageManager packageManager = applicationContext.getPackageManager();
        if (null != packageManager) {

            final String strPackageName = "com.attidomobile.passwallet";

            Intent startIntent = new Intent();
            startIntent.setFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
            startIntent.setAction(Intent.ACTION_VIEW);

            Intent passWalletLaunchIntent = packageManager.getLaunchIntentForPackage(strPackageName);
            if (null == passWalletLaunchIntent) {

                // PassWallet isn't installed, open Google Play:
                if (launchGooglePlay) {

                    String strReferrer = "";

                    try {
                        final String strEncodedURL = URLEncoder.encode(uri.toString(), "UTF-8");
                        strReferrer = "&referrer=" + strEncodedURL;
                    }
                    catch (UnsupportedEncodingException e) {
                        e.printStackTrace();
                        strReferrer = "";
                    }

                    try {
                        startIntent.setData(Uri.parse("market://details?id=" + strPackageName + strReferrer));
                        applicationContext.startActivity(startIntent);
                    }
                    catch (android.content.ActivityNotFoundException anfe) {
                        // Google Play not installed, open via website
                        startIntent.setData(Uri.parse("http://play.google.com/store/apps/details?id=" + strPackageName + strReferrer));
                        applicationContext.startActivity(startIntent);
                    }
                }

            }
            else {

                final String strClassName = "com.attidomobile.passwallet.activity.TicketDetailActivity";

                startIntent.setClassName(strPackageName, strClassName);
                startIntent.addCategory(Intent.CATEGORY_BROWSABLE);
                startIntent.setDataAndType(uri, "application/vnd.apple.pkpass");

                applicationContext.startActivity(startIntent);

                return true;
            }
        }
    }
    return false;
}
```

And an example call is:

```
launchPassWallet(getApplicationContext(),
Uri.parse("http://test.attidomobile.com/PassWallet/Passes/AttidoMobile.pkpass"), true);
```

Registered Company Number: 06428260

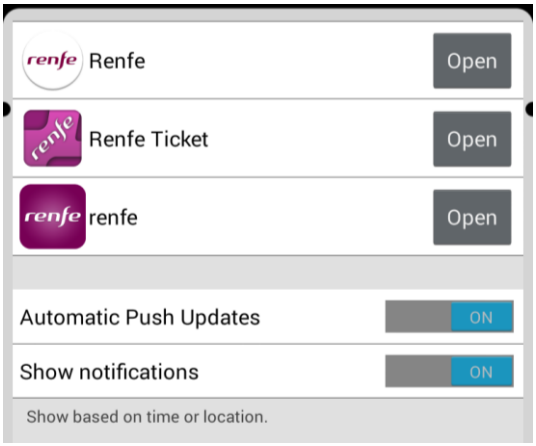
Registered VAT Number: GB 930 4201 71

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Linking Android Apps to a Pass

As of v1.31 of PassWallet, support for linking a pass to an Android app was added.

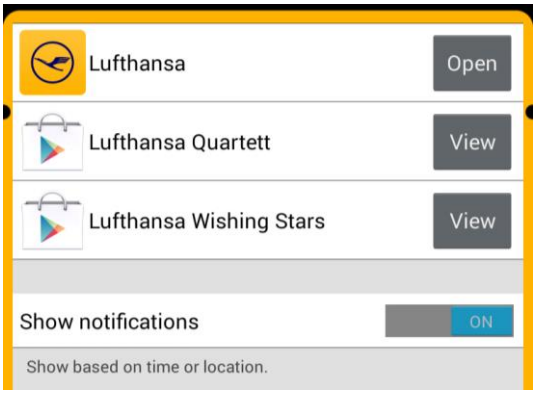


In order to link the pass, add the following section to the pass.json as a top level field:

```
"associatedApps":[
   {
      "title":"PassWallet - Passbook + NFC",

   "idGooglePlay":"com.attidomobile.passwallet",
      "idAmazon":"com.attidomobile.passwallet"
   }
]
```

One or more apps can be associated with the pass.



Individual package names can be specified for Google Play and Amazon App Store to handle there being platform specific variations of the app.

If the app is not already installed on the device then the specified “title” text will be displayed along with the icon of the appropriate app store (based on the app store that PassWallet is installed from).

If PassWallet is installed from a source other than Google Play or the Amazon App Store then no links to download the app will be displayed, only links to open apps that are already installed.

Registered Company Number: 06428260

Registered VAT Number: GB 930 4201 71

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Linking to passes guidelines

We also have various graphics that match the Passbook images for adding passes, to indicate to non-iOS users that they can actually use the passes, and link to download the viewer.

Please refer to https://q.pass.is/Amqm9SN7yzAB from an iOS & Android device as an example of their use.

The images we created we thought would be used for two scenarios:

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Downloading PassWallet

Consists of an image to indicate to the user they can download the app and a link to download from the correct app store. e.g.

```
<a href="http://passwallet.attidomobile.com/Download2.php">
 <img src="http://passwallet.attidomobile.com/images/download_image_100x136_dark_Viewer.png">
</a>
```

You then have a choice of wording & sizes for the image to use:

- http://passwallet.attidomobile.com/images/download_image_100x136_dark_PassReader.png
- http://passwallet.attidomobile.com/images/download_image_100x136_dark_Reader.png
- http://passwallet.attidomobile.com/images/download_image_100x136_dark_Viewer.png
- http://passwallet.attidomobile.com/images/download_image_212x100_dark_PassReader.png
- http://passwallet.attidomobile.com/images/download_image_212x100_dark_Reader.png
- http://passwallet.attidomobile.com/images/download_image_212x100_dark_Viewer.png

We can create other sizes / language specific versions of these on request, just email passwallet@attidomobile.com with the size & text required.

The "Download2" script looks at the user agent of the device and will redirect to the correct app store if on a supported device. We currently support:

- BlackBerry --> http://appworld.blackberry.com/webstore/content/137798/
- Kindle Fires --> http://www.amazon.com/Attido-Mobile-PassWallet/dp/B0091TL3TI
- Android --> https://play.google.com/store/apps/details?id=com.attidomobile.passwallet
- all other devices --> http://passwallet.attidomobile.com/Download.html --> Attido's default landing page

You can specify your own landing page for unsupported platforms like this:

- http://passwallet.attidomobile.com/download.php?default=http://www.google.com

You can also specify the URL of the pass, and when PassWallet is first opened (after installation) it will be imported automatically.

This can be done using the “pass” argument, e.g.:

http://passwallet.attidomobile.com/download2.php?pass=http://test.attidomobile.com/PassWallet/Passes/AttidoMobile.pkpass

or

http://passwallet.attidomobile.com/download2.php?default=http://www.google.com&pass=http://test.attidomobile.com/PassWallet/Passes/AttidoMobile.pkpass

Replacing http://test.attidomobile.com/PassWallet/Passes/AttidoMobile.pkpass with the URL of the pass.

ATTIDO MOBILE

Greenheys building
Manchester Science Park
Manchester
M15 6JJ

Tel: 0161 4089342 Web: www.attidomobile.com

Opening/downloading passes

Currently websites are using the "Add to Passbook" buttons supplied by Apple. This is potentially confusing for people who don't have Passbook.

So we created a link that will redirect to the correct button for that platform.

The links are:

http://passwallet.attidomobile.com/GetAddToButton.php?size=244x80

- Android/BlackBerry/Kindle --> http://passwallet.attidomobile.com/images/Add_to_PassWallet_244x80.png
- all other platforms --> http://passwallet.attidomobile.com/images/Add_to_Passbook_244x80.png

or

http://passwallet.attidomobile.com/GetAddToButton.php?size=122x40

- Android/BlackBerry/Kindle --> http://passwallet.attidomobile.com/images/Add_to_PassWallet_122x40.png
- all other platforms --> http://passwallet.attidomobile.com/images/Add_to_Passbook_122x40.png

URL Naming

In order to have your passes automatically opened in PassWallet, rather than downloaded to the phone's Downloads app/folder, the link must be identifiable as a pkpass file.

In order to achieve this with dynamically created passes, we use the following method:

On our server we have a PHP script (getPass2.pkpass.php), and we have the server setup to not require the file extension (so we can hide the tech used to implement the end point).

So:

http://test.attidomobile.com/PassWallet/getPass/getPass2.pkpass.php?number=0000

can also be accessed as:

http://test.attidomobile.com/PassWallet/getPass/getPass2.pkpass?number=0000

The latter one, as it appears to end in .pkpass, is automatically loaded in PassWallet when the link is clicked.