Write Your First ownCloud App in 60 Minutes

ownCloud makes it easy to securely store, sync, and share your files and documents, and access them from all your devices and platforms. But ownCloud is even more than a modern file sync and share solution. Like apps on a smart phone, ownCloud supports independently created applications as plug-ins or extensions that can run on top of ownCloud and provide additional features and functionalities.

Example of apps that have already been created for ownCloud include media player, calendar, and contacts. ownCloud is a platform for a wide range of HTML5 applications and web services that you can host and run under your own control.

Best of all, if you have some basic programming skills in a few common languages, you can create your own ownCloud app in about 60 minutes. Why develop an app for ownCloud? ownCloud is an open platform, and that means everybody is welcome to join the ownCloud core developer community to help improve the core features and push ownCloud forward. But even if you don't want to do this, there are reasons why you should consider writing an ownCloud application. Perhaps you are missing a feature and want to extend ownCloud to support it. Perhaps you don't want to rely on proprietary services like Gmail for your web mail interface, or you want your own application to manage your DVD collection.

You can easily write your own ownCloud application to implement any of these features, using an HTML5 interface or an API to provide the web service you want.

Many people write ownCloud applications to build a syncing and sharing backend for an existing application. Some examples are:

- You want to extend a photo management desktop application with the possibility to share photos with others.
- You want to extend your game to store a shared high-score list in a central place.
- You want to extend your notes application or RSS reader to sync data and status between different devices.

How easy is it?

Writing ownCloud applications is easy. You need some basic PHP and HTML knowledge to write your first application. Depending on the type of application you want to build, some JavaScript and SQL knowledge is also useful. ownCloud has a very open and friendly community so you can always get help from experts via mailing lists, IRC chat channels, or the forum at **https://vimeo.com/81178032**.

What do you need to know?

A working knowledge of the following programming environments is needed to write an ownCloud app.

PHP – You should know how to code some basic PHP and have a basic understanding of how the web and web servers work. The PHP syntax is similar to C or C++ and a lot of other languages, so it should be easy to learn. ownCloud provides a powerful framework to help you with things like user management and templating.

HTML – You need some basic HTML and CSS knowledge to develop the user interface for your application. ownCloud provides functionality to create configuration dialogs and other basic user interface elements. ownCloud uses jQuery to enable the creation of a rich HTML5 web interface as part of your application.

SQL – ownCloud has functionality to read and write basic data to the database or file system. You will need some SQL knowledge if you want to work with more complex data structures in your application. This is optional and depends on the type of application you want to create.

ownCloud API – The ownCloud core proviwdes a stable public API for ownCloud application developers. You

About ownCloud

ownCloud is a software platform that can replace proprietary cloud services such as Google Drive, Dropbox or box.net. Users can host an ownCloud installation on their own hardware or at a provider that offers ownCloud hosting. ownCloud can also run on most basic web spaces that support PHP without the need for special configurations. ownCloud is suitable for home users and big organizations alike, and is available under the AGPL license as free software. It is developed by an open, friendly and worldwide community of developers. Professional enterprise support and services for ownCloud are delivered by ownCloud, Inc. Windows, and Linux will need to be familiar with the basic calls that are required for registering your application, loading and saving configurations, and other basic functionalities.

How to start

A good place to start is to look at the ownCloud apps already written. You can find all the ownCloud code in the apps repository at **https://github.com/ownCloud**, a leading site for open source code sharing. You’ll find many applications here that will help you learn how ownCloud apps are structured. You may also find apps similar to the ones you are considering, and you may want to contribute to those before writing your own.

The application template provided at **https://github.com/owncloud/apps/tree/master/appframework** is a good starting point for development. You can always ask questions on the mailing lists, forum, or the chat channel if you need help. You can find all the possible ways to contact the core team here: **http://owncloud.org/contact/**.

The directory structure

ownCloud applications have to be structured in a specific way to work. The recommended directory structure is:

- **ajax** – a directory for the ajax call controller files if needed in your application.
- **appinfo** – The `info.xml` and the `app.php` are located in this folder. The most important file is app.php. ownCloud will call this file every time it runs, so this is the place where you can register your navigation entries or connect signals and slots. You can also put a database.xml into this folder if needed.
- **css** – The additional stylesheets of this application are located here.
- **img** – All the needed images should be located in this folder.
- **js** – You can put you Javascript files here. The JS libraries we use are jQuery and Torch. You can add scripts to your pages using `OCP\Util::addscript`. Scripts can be added this way in either `appinfo/app.php` or in your app‘s individual PHP scripts.
- **l10n** – Put your translations into this folder if you want to support more than English. As soon as the translation detects a l10n folder it knows that this program is translatable. You can find more information about the translation system here **http://owncloud.org/dev/translation/.**
- **lib** – Application logic, classes, and functions go into php files in this folder. ownCloud uses the MVC principle. The libraries represent the model. Basically all data should be handled here so the php files that the user calls only interact with the library. If you have a look at the lib folder in, for example, the contacts app, you will see three files: addressbook.php, hooks.php and connector_sabre.php. The addressbook.php file contains all methods that are required for handling addresses. We mostly use the classes as namespaces and write static functions; “real” OOP with dynamically instansiated objects is only used if needed.
- **templates** – Templates for your user interface should go into this folder. ownCloud has its own template system. The templates are php files that are included by OC Template. Having a look at an example file should show you everything you need to know. The variables are assigned by using `$tmpl->assign(‘name’, ’value’);`. You can find more information here: **http://owncloud.org/dev/apps/getting-started/**.
- **The API** – You can find the complete API documentation here: **http://owncloud.org/dev/apps/getting-started/**. The most important calls are:

`OCP\App::addNavigationEntry`
This adds an navigation of your application to the main navigation on the left. You usually call this in `app.php`.

`OCP\App::setActiveNavigationEntry`
This calls activated a navigation entry. You do this when inside your application.

`OCP\App::registerPersonal`
You can use this call to register an personal settings section on the user settings page.

`OCP\App::registerAdmin`
You can use this call to register an admin settings section on the main admin page.

`OCP\Config::getUserValue`
Use this call to read a setting for the

Helpful links

Getting started:
http://owncloud.org/dev/apps/getting-started/

Existing apps:
https://github.com/ownCloud

Application template:
https://github.com/owncloud/apps/tree/master/appframework

Contacting the core team:
http://owncloud.org/contact/

currently logged in user from the settings store.

`OCP\Config::setUserValue`
Use this call to write a setting for the currently logged in user to the settings store.

`OCP\DB::prepare`
Prepare an SQL statement for execution.

`OCP\User::isLoggedIn`
Check if a user is currently logged in.

`OCP\Util::addscript`
Load an additional JS script

`OCP\Util::addStyle`
Load an additional CSS style

Conclusion

One of the great advantages of working with an open platform like ownCloud is its easy extensibility. There is a whole community of users ready to help you get started. If you have an idea for a service to run on top of ownCloud, consider developing it yourself. You'll find it's easy, fast, and well worth the effort.

US Headquarters
ownCloud, Inc.
57 Bedford Street, Suite 102
Lexington, MA 02420
United States

www.owncloud.com/contact

European Headquarters
ownCloud GmbH
Schloßäckerstraße 26a
90443 Nürnberg
Germany

www.owncloud.com/de/contact