Table of Contents

Getting started

Tutorials

TagSpaces Extensions

Appendix Links

Introduction

This is the first draft of documentation for TagSpaces products as of **version 3**. Please note, that there are still many places where we show content and screenshot from the old version 2 of the app.

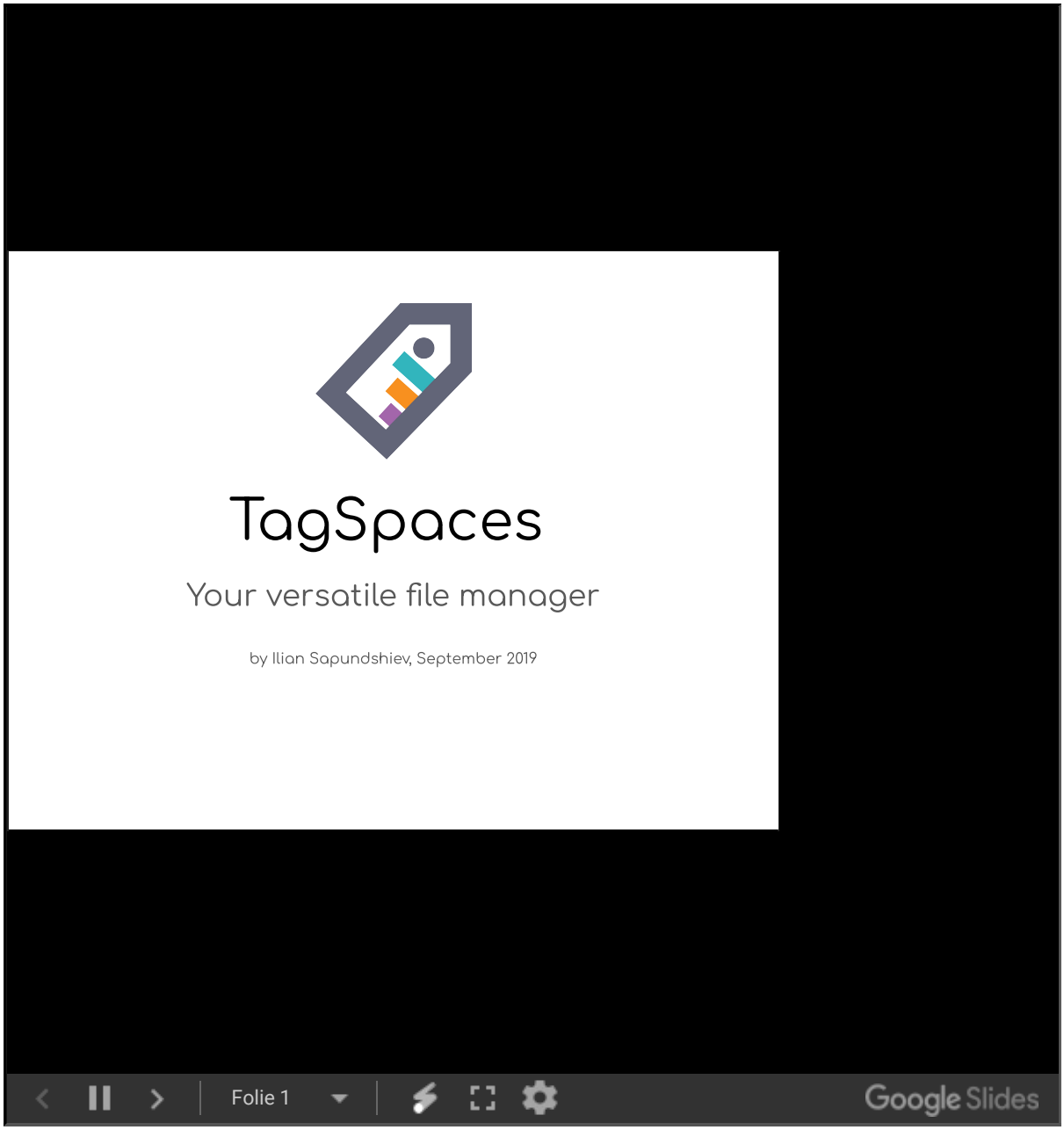
> **Warning:** This documentation is in **work in progress** status!

About TagSpaces

TagSpaces is an open source, cross-platform, no backend, no login, personal data manager, and file navigator. It helps you organize your files, photos and other documents with tags on different platforms and/or devices, according to your own preferences and following your own logic, by using the same smart and friendly interface everywhere. The short introductory video below shows TagSpaces in action:

Check out the TagSpaces YouTube channel for more TagSpaces related videos.

The following few slides explain the basics of the project. To navigate the presentation use the arrow keys or click on it and use the arrow key on your keyboard.



Products landscape

- **Community desktop version** - the main, community developed edition of TagSpaces
- **PRO desktop version** - extended solution based on the community version for advanced users
- **Web Clipper as Firefox addon** - a web clipper allowing the saving of whole webpages, webpage fragments and screenshot as local files
- **Web Clipper as Chrome extension** - same as the web clipper for Firefox
- **Android app** - an app offering most of the features from the community desktop version on a Android device (currently in development)
- **Web self hosting edition** - A package for self hosting on top of a any WebDAV server such as Nextcloud or ownCloud.

Downloads

All versions of TagSpaces are available to download from the Downloads Page, where you can quickly find the appropriate installer for your platform (Windows, Mac, Linux, Android, Firefox and Chrome).

About this document

> **Hint:** This documentation has currently a **WIP status**. The articles in this documents are being rapidly updated, with new information added regularly.

The structure of this documentation

This documentation is built using GitBook, for a streamlined and simple user experience. Each **page** will concern a particular topic, giving **detailed and illustrated explanations** and **instructions** about it. Every page is broken down to different level headings:

Section titles

...will divide the chapters into smaller chunks

Subsection titles

...will divide sections, while

Smaller headings

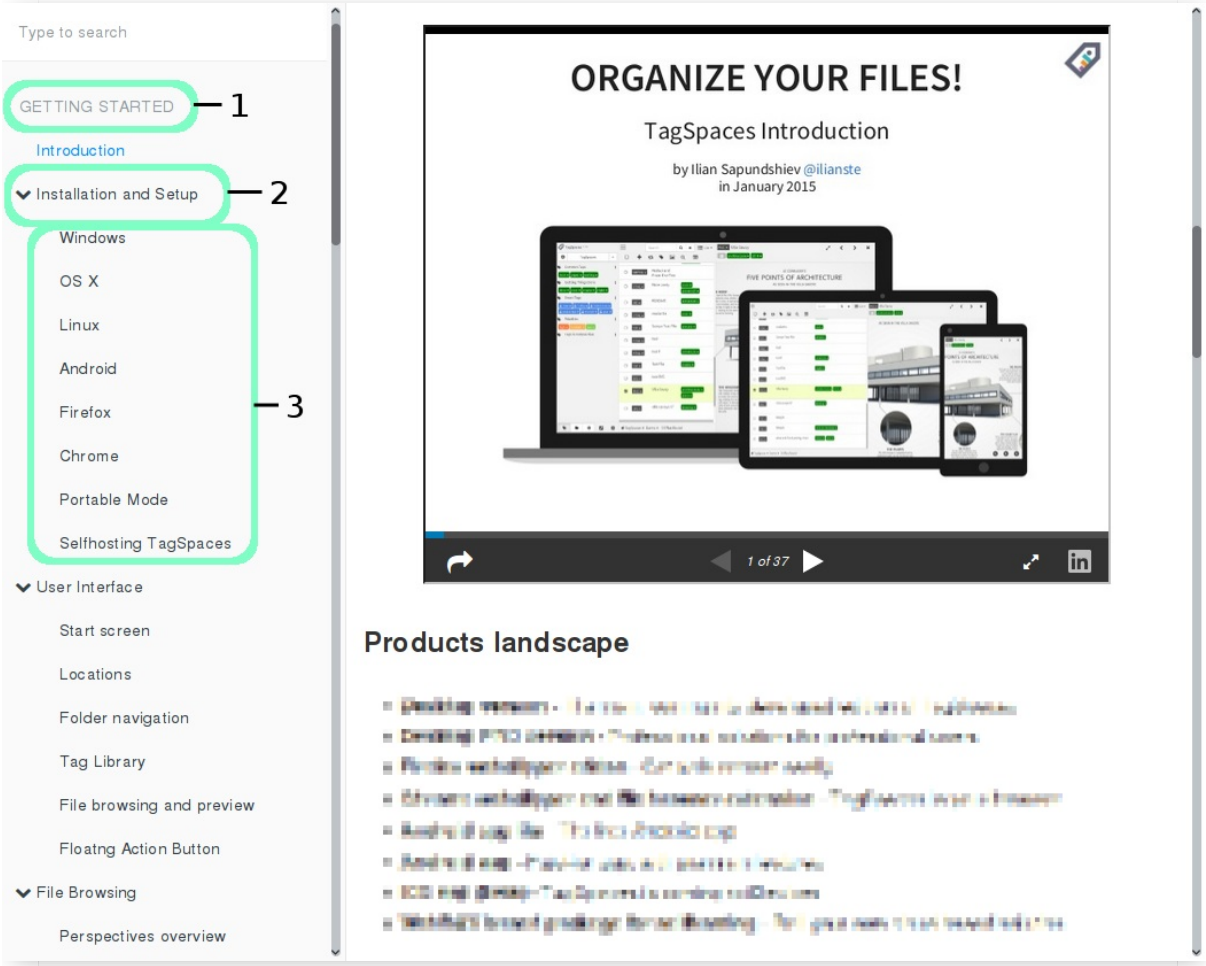
...might be included occasionally, for topics with specific significance

---

The navigation sidebar on the left side of this page will represent the **chapters** as expandable topics, with the **sections** listed in each expanded view.

Illustrations

Much effort had gone into visually illustrating the documentation, so that you can instantly find what you are looking for, or what is being described in words. Most illustrations will feature some sort of annotation. For example if a **visible element** of the User Interface is discussed, a **neon-green rectangle around an element** will mark its location on the illustrating image.



When there are multiple elements discussed in the same paragraph, the illustrations will be further annotated with numbers, like on the image below while the corresponding numbers will be included in the text, after each element was first mentioned. For example the following image illustrates the structure of the navigation sidebar, with **major parts** (**1**) showing as unselectable titles, expandable **page titles** (**2**) will hold major **section titles** (**3**), both of which are clickable. Clicking page titles will open the corresponding page, while clicking section titles will open the page and scroll down to the start of the selected section.

> **Note to contributors:** When you annotate illustrations please use a rounded edge rectangle, with the following properties:
>
> - **20 px radius** for the rounded corners
> - **#1ccd9c color**
> - **3-6px line width**, depending on image size (larger images can use thicker lines) Additionally, please use a **28pt font size** for numbering with a basic **sans font**.

Text markup

You will notice, that certain words are marked with **bold text**. These either mean names of elements, or significant notes/concepts about usage. *Italicized words* usually mark menu items, or other selectable elements, although it is not a hard and fast rule.

Bulleted lists will be used to

- Improve **readability**
- Make it **easier to find** what you are looking for.

Document symbols

There are currently two types of symbols, apart from the usual text formatting and annotations, that you can find on these pages:

- pro - means that the described feature is part of the TagSpaces Pro and Enterprise editions.
- ⚒ - means that the section is not ready yet and may contain unclear, or not up-to-date information, or sections might be missing entirely.

Contribution

This documentation project is hosted on GitHub, and uses the GitBook format. Enhancement, or corrections are welcome via pull requests. For the markdown syntax used for the document please refer to the GitBook markdown page

> **Note:** If you are planning to contribute, please do focus on articles marked with the ⚒ symbols (marking **work in progress**) initially.

Credits

Original text and images in this documentation were **created and edited by**:

- **Attila Orosz** - main editor of the documentation for version 2 of the product, you can contact him via email.
- **Ilian Sapundshiev** - initiator of the TagSpaces project

Articles published on the TagSpaces Blog, served as the **basis upon which this documentation was built**. Fragments of the original text can still be found in the documentation, without marking the original author.

License of the documentation

Installation and Setup

Installing Windows version

The Windows version of TagSpaces is distributed as `zip` file, which can be downloaded for 64bit versions of Windows, depending on your system. Then you have to just unzip the downloaded file in a folder of your choice somewhere on your system. In the unpacked folder you will find a file called `tagspaces.exe`, which can be executed with double click.

Updating the Windows version

Just exchange the content of your current installation, with the content from the zip-file of a newer release. You don't have to worry about your tag library and locations since they are persisted in the users home folder.

Installing macOS version

For macOS 10.10 or later you have to download the 64bit of the application. Then you have to double click the downloaded zip file in Finder to unpack it properly. Some users are reporting issues with unpacking the app with some 3th-party zip utils.

> **Note**: By default macOS does not allow TagSpaces to be executed with a simple double click, because it is currently not digitally signed. To work around this, right-click the TagSpaces.app folder and choose Open, then click **Open** on the dialog that appears. You need to do this just once.

Updating the macOS version

Just exchange the content of your current installation, with the content from the zip-file of a newer release. You don't have to worry about your tag library and locations since they are persisted in the users home folder.

Installing Linux version

The Linux version of TagSpaces is distributed as `tar.gz` file, which can be downloaded for 64bit Linux versions, depending on your system. Then you have to just unpack the downloaded file in a folder of your choice somewhere on your system. In the unpacked folder you will find a file called `tagspaces`, which can be executed with double click or in a terminal. We also offer 64bit deb package for Debian/Ubuntu based distributions.

Updating the Linux version

Just exchange the content of your current installation, with the content from the tar.gz-file of a newer release. You don't have to worry about your tag library and locations since they are persisted in the users home folder.

Installing Android version

You can install the Android app directly from the Google Play Store.

Updating the Android app

Once we publish a newer version of the Android app, it will be automatically installed on your mobile device within the

Once we publish a newer version of the Android app, it will be automatically installed on your mobile device within the following days.

Installing the Firefox Addon

Download the Add-on for Firefox. Open the **Add-on tab** in Firefox from the main menu and then click on the **Add-on** puzzle icon, or simply type `ctrl+shift+a` or `command+shift+a`. In the opened new tab, go to the settings dropdown on the left of the search field and select **Install Add-on From File...**. Then choose the downloaded file and follow the instructions.

You can also find TagSpaces in the Mozilla's addon store, but consider that the version there could be some months old due to their slow approval process.

Once installed the add-on can be started from the TagSpaces icon in the right upped corner of the browser or from the browser's main menu **Tools > TagSpaces**.

Updating the Firefox addon

Once a newer version of the Firefox addon is approved for the Mozilla store, it will be automatically installed on your Firefox browser within the following days.

Installing the Chrome Extension

You can install the Chrome extension directly from the Google Chrome Store.

Installation from the TagSpaces webpage

An alternative way is get the extension directly from our download page. If you do so, please consider the following instructions:

- Download the extension file from the link above and save it to your computer.
- Click the settings icon on the browser toolbar.
- Select **Tools > Extensions**.
- Locate the extension file on your computer and drag the file onto the Extensions page.
- Review the list of permissions in the dialog that appears and click **Install**.

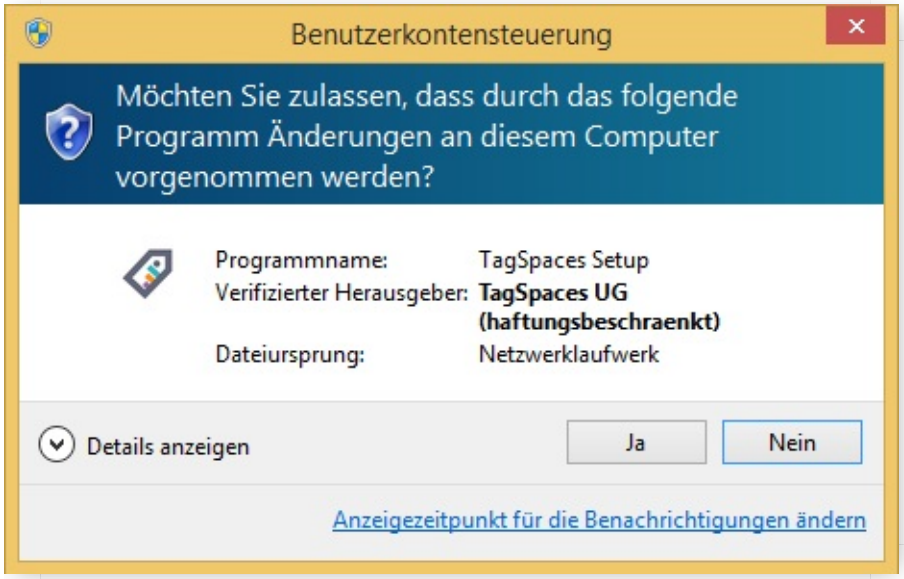
After the installation, the TagSpaces extension can be started from the TagSpaces icon in the right upper corner of the browser.

Updating the Chrome extension

Once we publish a newer version of the Chrome extensions on the Chrome store, it will be automatically installed on your Chrome browser within the following days.

Installing TagSpaces PRO

For Windows we are providing installers for TagSpaces PRO, which are signed with modern software certificates. The following screenshot show this certificate in action during the installation under Windows.



Here are the steps needed for installing the PRO version on your system.

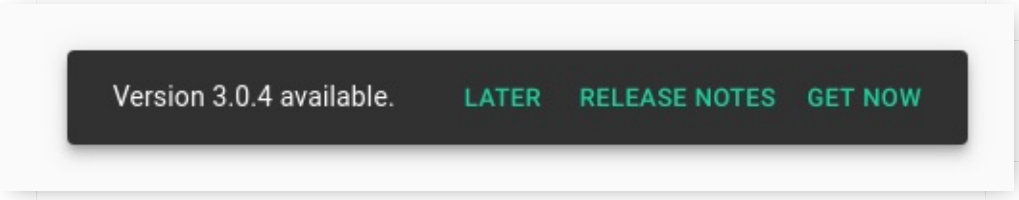
- Save the installer file to a location on your hard drive.
- Locate the installer and double-click to start the install process.
- The Windows installer allows you to specify a custom install location.

Updating the PRO version

Just run the installer from a newer version of TagSpaces. It will automatically update your installation.

New version notification

Once a new version of the application is available, you will see a green present icon near the TagSpaces logo. Clicking on this icon will lead your to the what's new webpage on the TagSpaces website. Here you can download the new version of the application.



If you do not want to TagSpaces to check for the availability of new version, disable this functionality in the settings

Running TagSpaces in portable mode

The Windows and Linux versions of TagSpaces can be started in the so called portable mode, allowing the use of the application from an USB-stick or other portable devices. In this mode all the configuration information such as tags, tag groups and connection to locations is stored in a folder called *tsprofile*. This folder is located in the from where the application is started and not in the home user folder on current operating systems. The portable mode can be started with the `tagspacesp.cmd` script under Windows or the `tagspacesp` under Linux. These scripts are located in the unzipped application folder. Under Linux you have eventually to make this script executable with this command `chmod +x tagspacesp` .

Updating the portable version

In order to update a version of TagSpaces used in portable mode for example on USB stick you have just to exchange the files and directories with the new ones, coming from the installation package of the new application version. As mentioned before in the portable mode, the tag library and locations are stored in the *tsprofile* folder. So to upgrade to a newer version of tagspaces, we have to take special care for this folder, please follow these steps for the upgrade:

- Rename your existing *tagspaces* folder to e.g. *tagspaces_old*
- Extract the downloaded tagspaces zip-file, this will create a new *tagspaces* folder.
- Copy/move the *tsprofile* folder from the *tagspaces_old* folder to the newly created *tagspaces* folder.

> **Important**: So once again be careful not to delete the `tsprofile` sub folder, it contains valuable information for your portable TagSpaces instance.

Self Hosting TagSpaces

Motivation

Almost since the very first releases of TagSpaces back in the 2013, many users did requested a server based version of TagSpaces. They wanted to use the convenient tagging workflow of TagSpaces on their self hosted Nextcloud/ownCloud or in general WebDAV instances. So starting from today this is possible, TagSpaces can now run on servers and once installed you can access your files from anywhere.

Download and Installation

The current version can of the WebDAV version can be downloaded from here.

> **Note** Please handle the current status of the implementation is a technology preview, which in our opinion is still not suitable for production use on Internet.

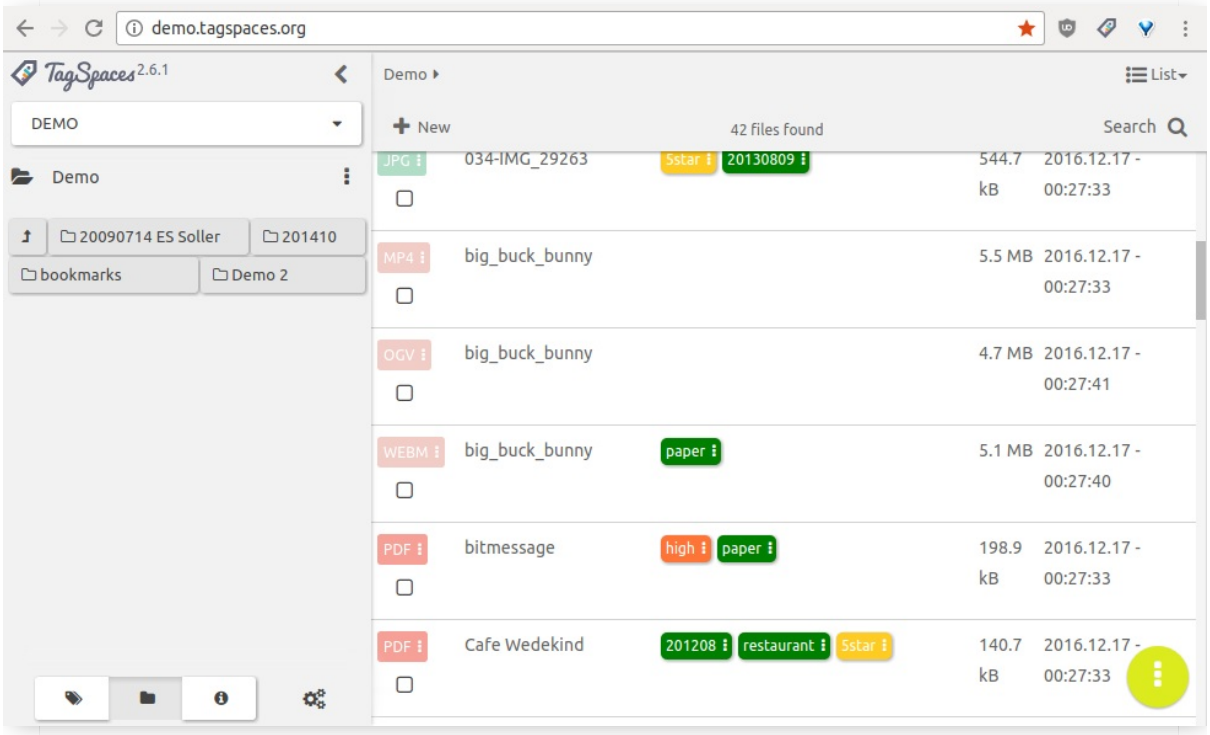
In order to use the hosted version you need a working WebDAV server. The current release was tested with success on ownCloud which is based itself on the sabre/dav WebDAV server. On Ubuntu the installation steps are as following:

- Install Apache webserver (*nginx* webserver with its webdav extension is reported also to work)
- Install Nextcloud/ownCloud or any other WebDAV server
- Unzip the hosted version of TagSpaces somewhere in the www root folder of Apache. Currently the TagSpaces should be on the same host/ip and using the same port as the WebDAV server. This is so because of the XSS prevention build in the modern internet browsers.
- Assuming *ownCloud* is installed in `/var/www/owncloud` and TagSpaces in `/var/www/tagspaces` you have to type something like this in your terminal:

```
cd /var/www
sudo chmod -R 755 tagspaces
sudo chown -R your_www_group:your_www_user tagspaces
```

- Open your browser and enter: `127.0.0.1/tagspaces`. The TagSpaces UI should be loaded.
- Create a new location with the following path: `/owncloud/remote.php/webdav`
- Give a name to your location and save.
- A dialog for credentials entering will appear. Enter here your ownCloud username and password.
- That's all, you can now browse your files in TagSpaces.

> Note: If the dialog prompting for the user credentials does not appear and you have running Nextcloud/ownCloud in some other browser tab, you have to logout there and reload the tab running TagSpaces.



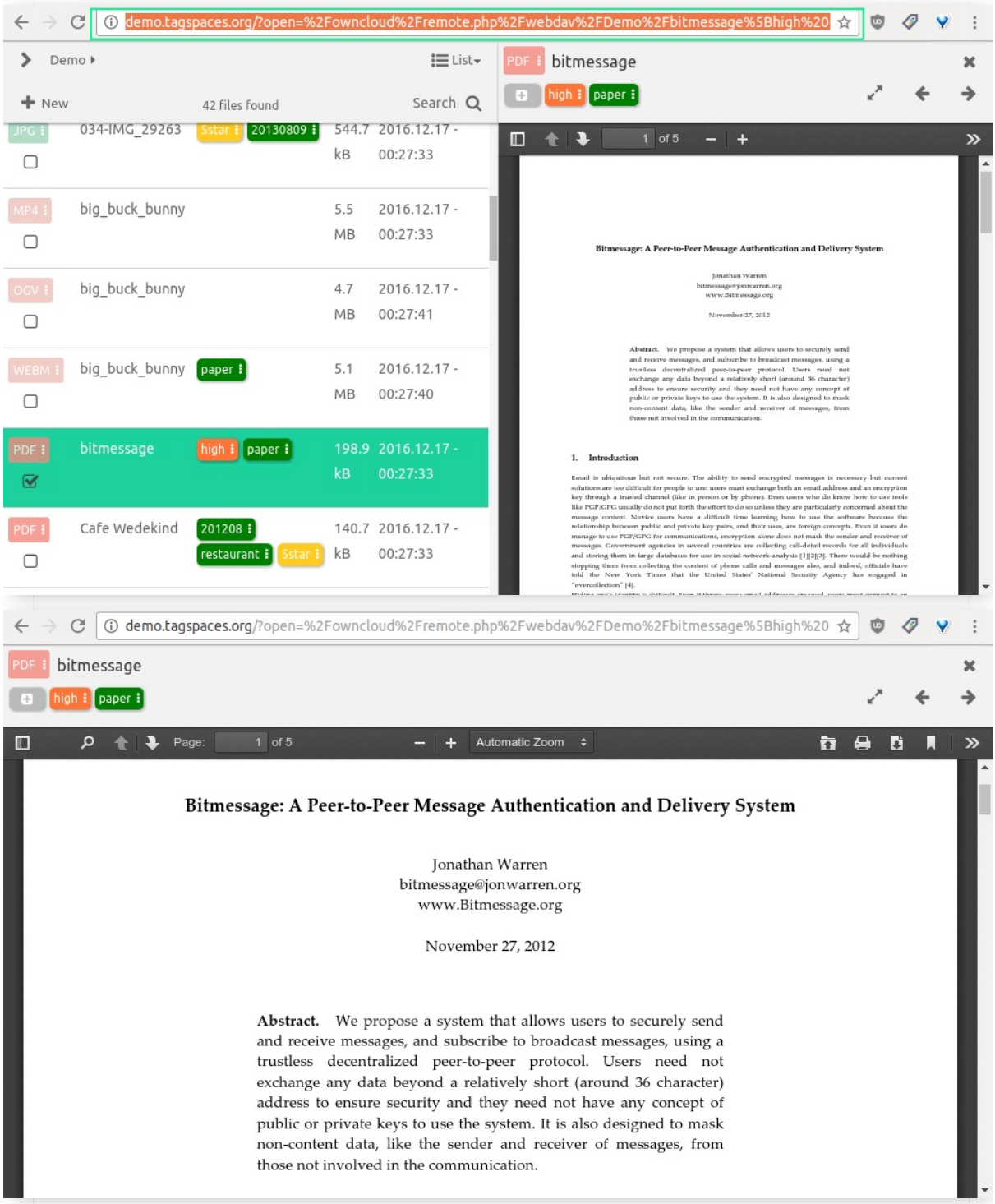
It is interesting to mention that you can open also your ownCloud contacts by creating a location with a path like `/owncloud/remote.php/carddav/addressbooks/ilian/contacts`. This does not make currently much sense, because you only see a list with VCF files, but who knows perhaps somebody will write a contacts perspective and VCF viewer for TagSpaces some day.

Demo

You can experience a live demo of the TagSpaces webdav version on demo.tagspaces.org. The username and password are both `demo`.

Sharing links to files from the webdav version

tbd



Starting the WebDAV edition locally for testing

There is a script called `webdavserver.js` located in the data/web , which can be started with:

```
node data/web/webdavserver.js
```

or

```
npm run webdav
```

This command will start a local node.js based WebDAV server on `http://127.0.0.1:8000`. Open your browser and enter the following URL:

```
http://127.0.0.1:8000/index.html
```

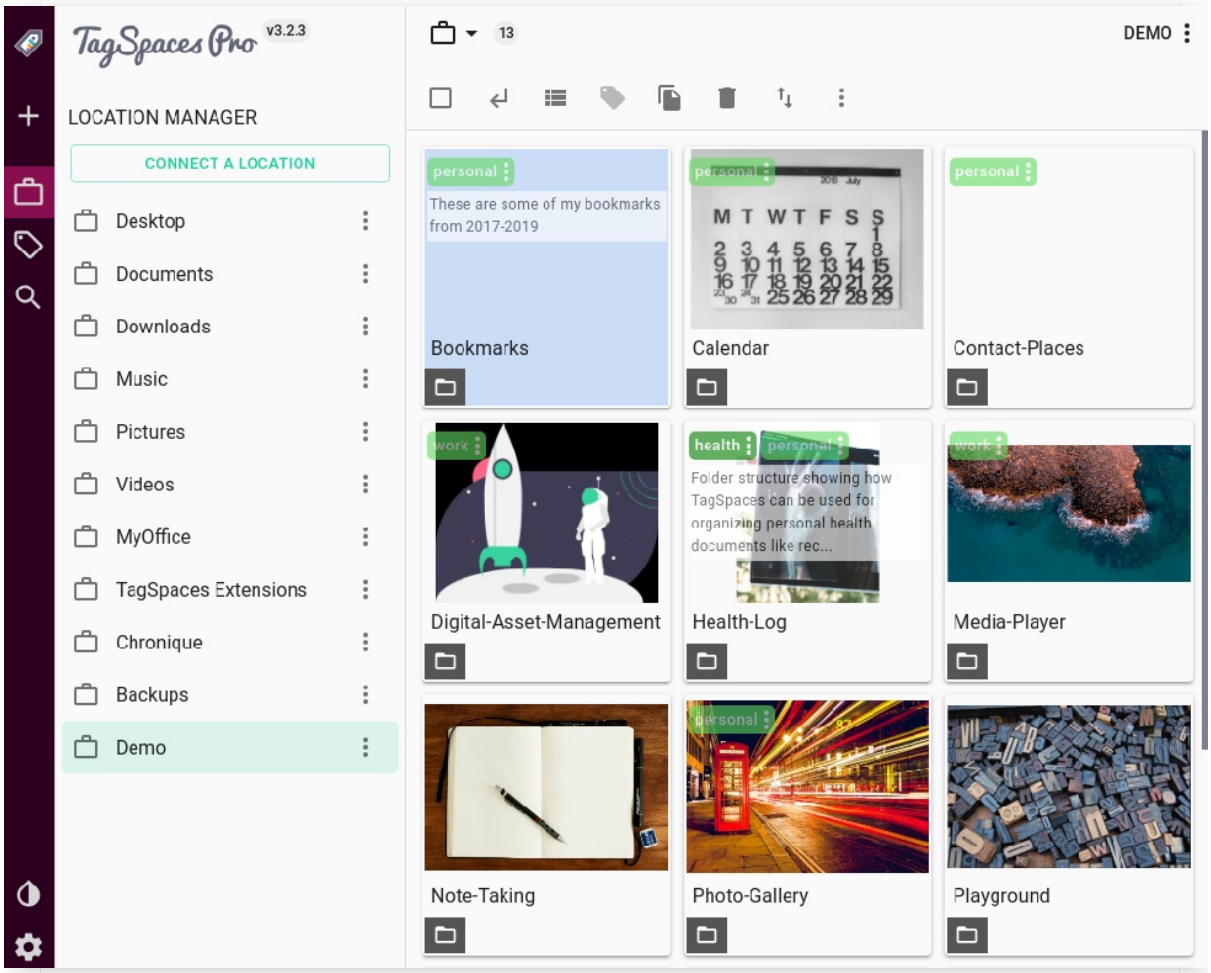
You will be prompted for user credentials, which are username: `demo` and password: `demo` and now you should be able to work with the WebDAV version of TagSpaces.

User Interface

Start screen

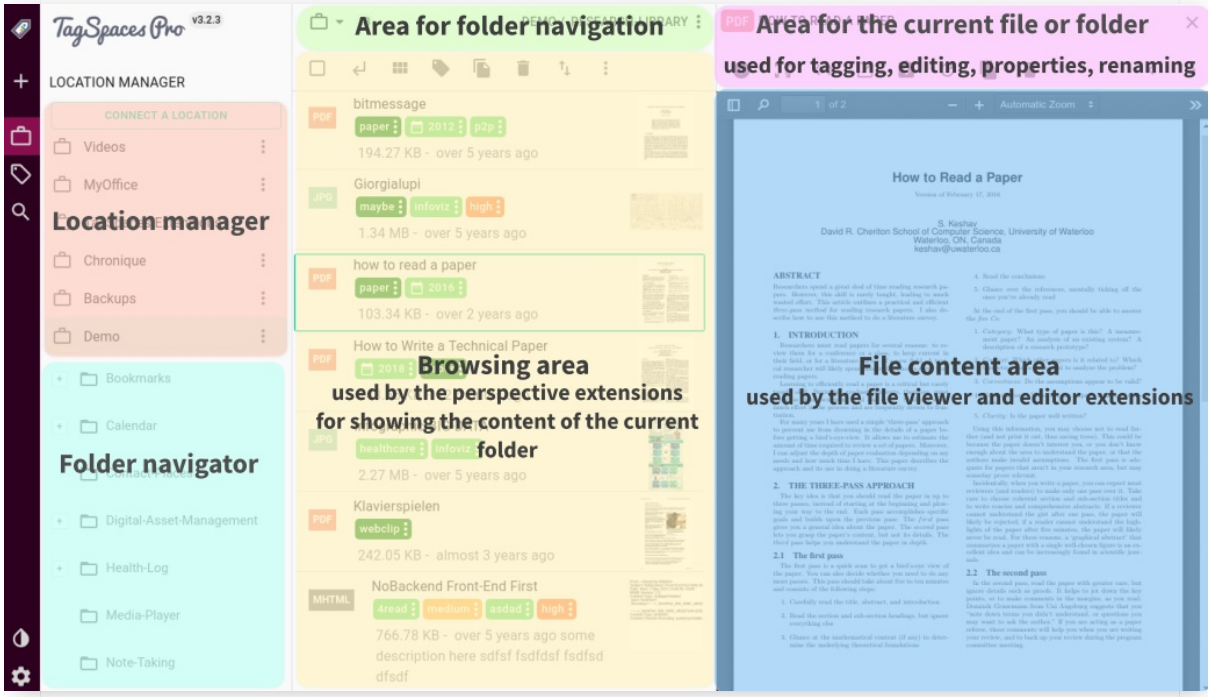
TagSpaces opens up as a single-window application that tries to make the best use of any space available, be it large wide screens, or small mobile displays. The user interface, devised along the guidelines of Google's material design specifications, is very adaptable. The application will look the same, regardless of the operating system, or platform it runs under.

This streamlined experience offers familiarity and an ease to use the application across platforms, while you will need to familiarize yourself with its extensive features only once. The following screen shows a typical start screen of TagSpaces with the list of the available locations on left, and content of the currently opened folder with the default perspective in center of the application.



The user interface of the application consists of the following main areas:

- Location management - located below the TagSpaces logo
- Folder navigator - located at the leftmost area of the user interface
- Tag library area - located at the leftmost area of the user interface (replaces the folder navigator, when active)
- File browser area - located in the main area of the user interface
- File content area - if a file is opened, the main area splits into two panes. The file viewer will occupy the right pane, while the left will be preserved for the file browser.



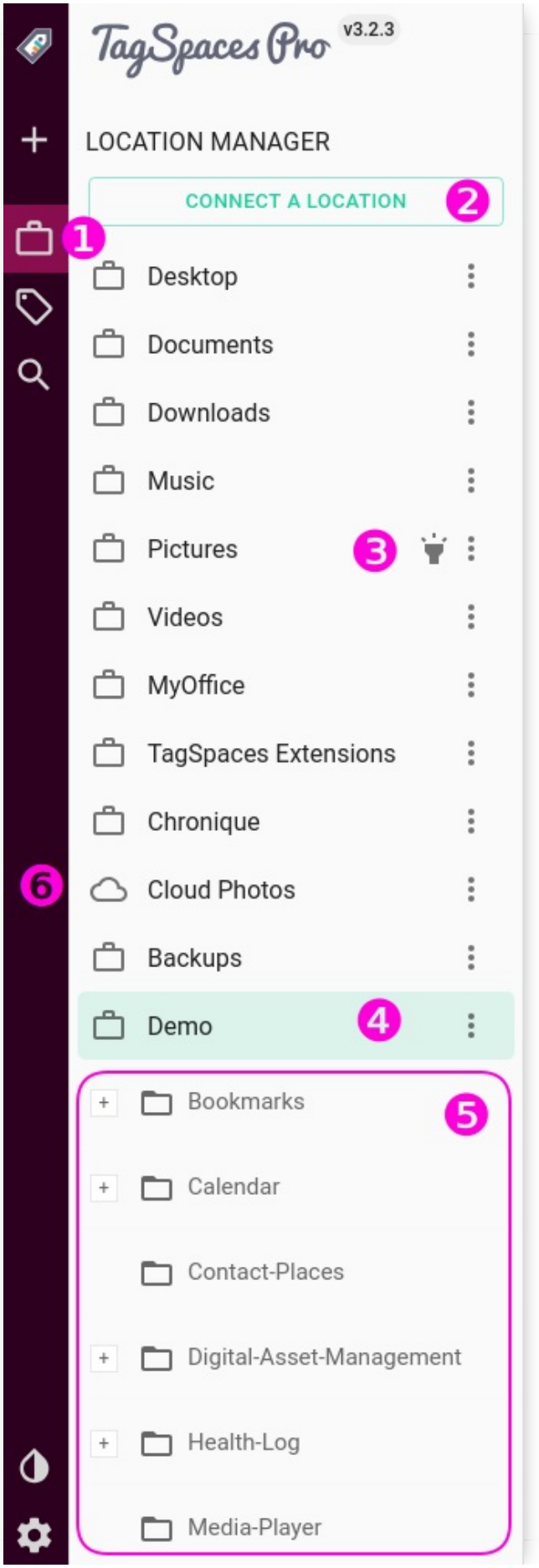
Locations

A **location** is a folder on your local file system, which will serve as a root for listing sub folders, files and documents. Typical locations are for example the folder where you collect your photos or folders where you store documents, ebooks or music.

> **Hint:** Do not choose very large directories such as your *home folder* as a location root folder, as this may lead to performance issues. The reason for this is that TagSpaces is indexing the whole location every time you open it and the indexing can just take time if the location contains many files. On modern computers with SSD harddrives having a location with up to 60000 files is usually not a problem. The current upper limit for files indexed in a locations is 200000.

The **location manager** is can be opened by clicking the case button (1) from the vertical navigation, or by pressing the *CTRL+1* key combination. Once opened, you will see a list of the currently configured locations. Here you can open any location by clicking its name. You can add new locations, by clicking on the **Connect new location** button (2). This button will open the *Connect a Location* dialog, which is very similar to the *Edit Location* dialog. The torch icon (3) indicates **Startup Location**, which is the location loaded automatically on the application start.

Clicking second time on a location name, will trigger the loading of the sub directories in this locations, once loaded this folder will be listed in the **folder navigation area** (5)


TagSpaces Pro
v3.2.3
LOCATION MANAGER
CONNECT A LOCATION
2
1
Desktop
Documents
Downloads
Music
Pictures
3
Videos
MyOffice
TagSpaces Extensions
Chronique
6
Cloud Photos
Backups
Demo
4
Bookmarks
5
Calendar
Contact-Places
Digital-Asset-Management
Health-Log
Media-Player

The location with the cloud icon (6) in front of its name is located in the AWS S3 compatible cloud storage. See cloud locations for more details.

The location with the light green background color (4) is the **currently opened location**. The 3-dot button located on the most right part of every location in the location manager will open the **context menu** for this location. This menu contains the following menu entries:

- *Edit Location* - will open the *Edit Location* dialog where you can change the location properties such as name or path.
- *Refresh Location Index* - will trigger the indexing process manually, this menu entry is visible only for the currently opened location
- *Move up* - will move the location visually up in the location manager
- *Move down* - will move the location visually down in the location manager
- *Remove location* - will remove the location permanently from the app. This operation will not affect your files, it only removes its reference in TagSpaces.
- *Show in File Manager* - will open the path in your file system to which this location point in the default file manager of your operating system
- *Close Location* - will simply close this location if it is currently opened.

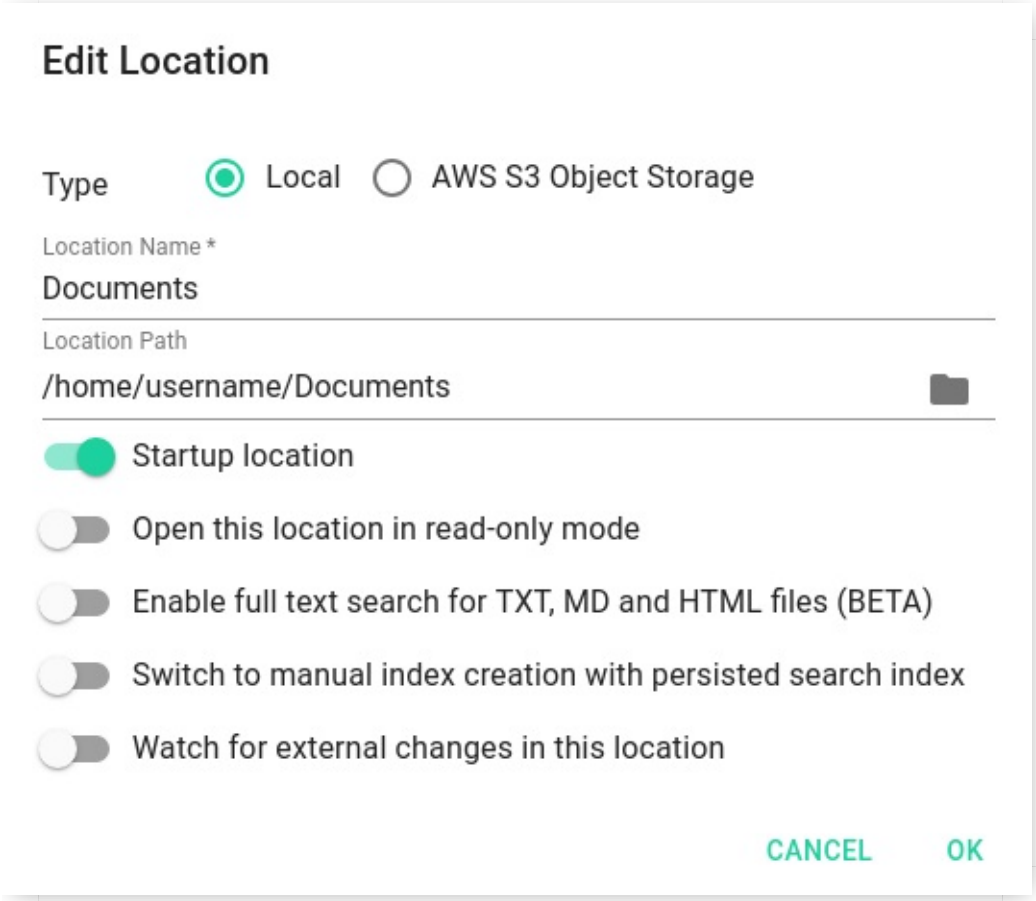
Location types

In TagSpaces there two type of locations, regular (local) and cloud based (AWS S3 Object Storage). The type can be selected in the create and edit location dialogs.

Regular Locations

Regular locations are pointing to a folder located on your local computer. This could be also a folder where you sync locally your Dropbox files or a folder from a connected network drive.

> **Note:** Be careful with connected huge folder located on network drive, depending on the speed of the network, this could lead to performance issues.



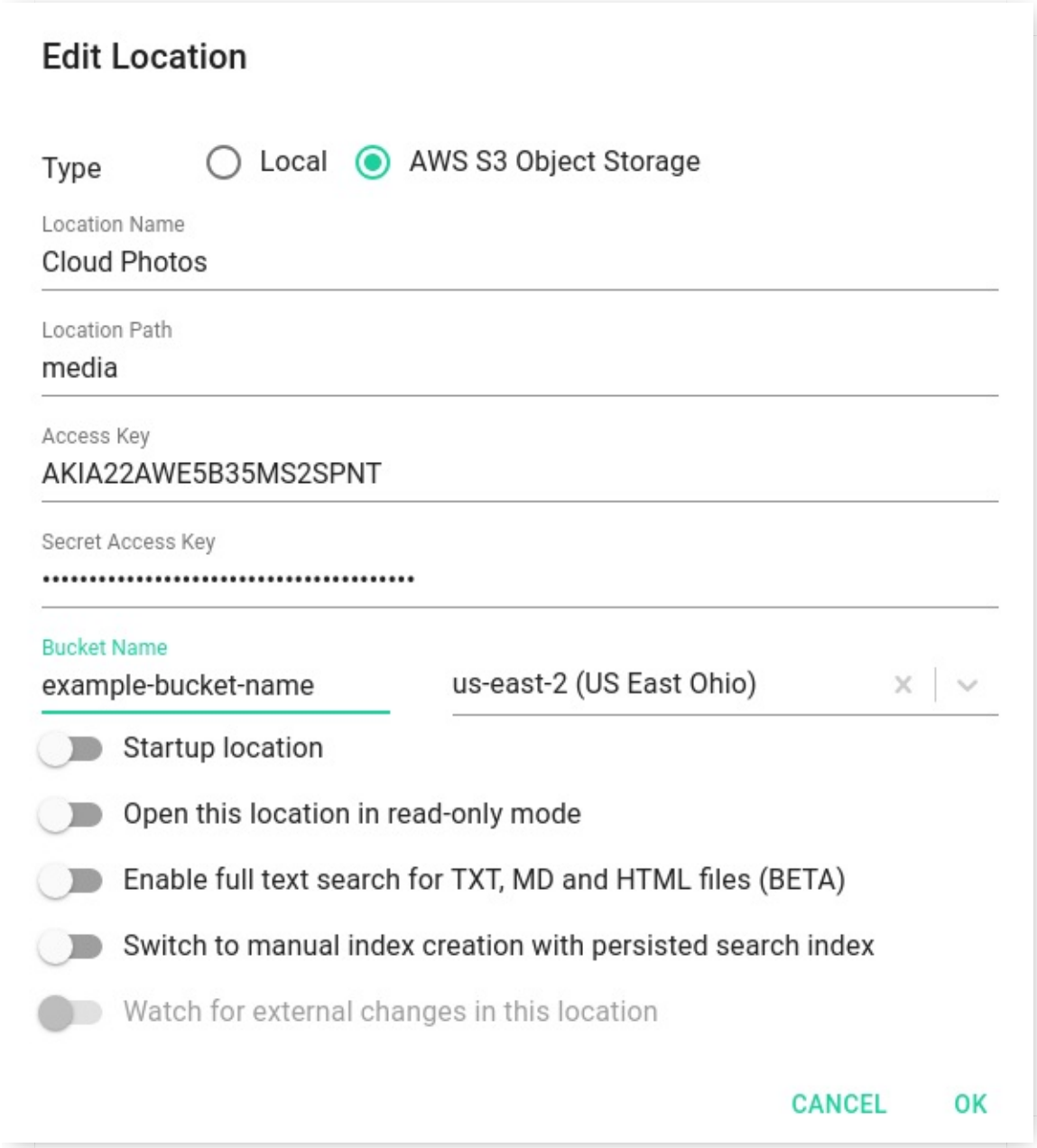
The regular locations have the following properties:

- *Location name* - this is the name of location as displayed in the location manager
- *Location path* - the path from your computer, to which this location points
- *Startup location* - turning this switch on will make the location load automatically on application start. You can set only one location to be a startup one.
- *Open this location in read-only mode* pro - switches the interface of TagSpaces to read-only mode
- *Enable full text search for TXT, MD and HTML files* pro - activates the indexing of the content of text, markdown and HTML files.
- *Switch to manual index creation with persisted search index* pro - disables the automatic indexing of a location on its opening. The application will try to open a previously created index located in a file *tsi.json* from the *.ts* folder of the location. This is useful on locations with many files, where the content does not change very often.
- *Watch for external changes in this location* pro - once switched on TagSpaces will watch the folder to which the location points and all its sub folders for changes and reflect them in the application.

Cloud Locations

These location are pointing to AWS S3 compatible stores (also known as buckets) located in the Internet.

> pro Cloud locations are available only in the Pro and Enterprise editions of TagSpaces.



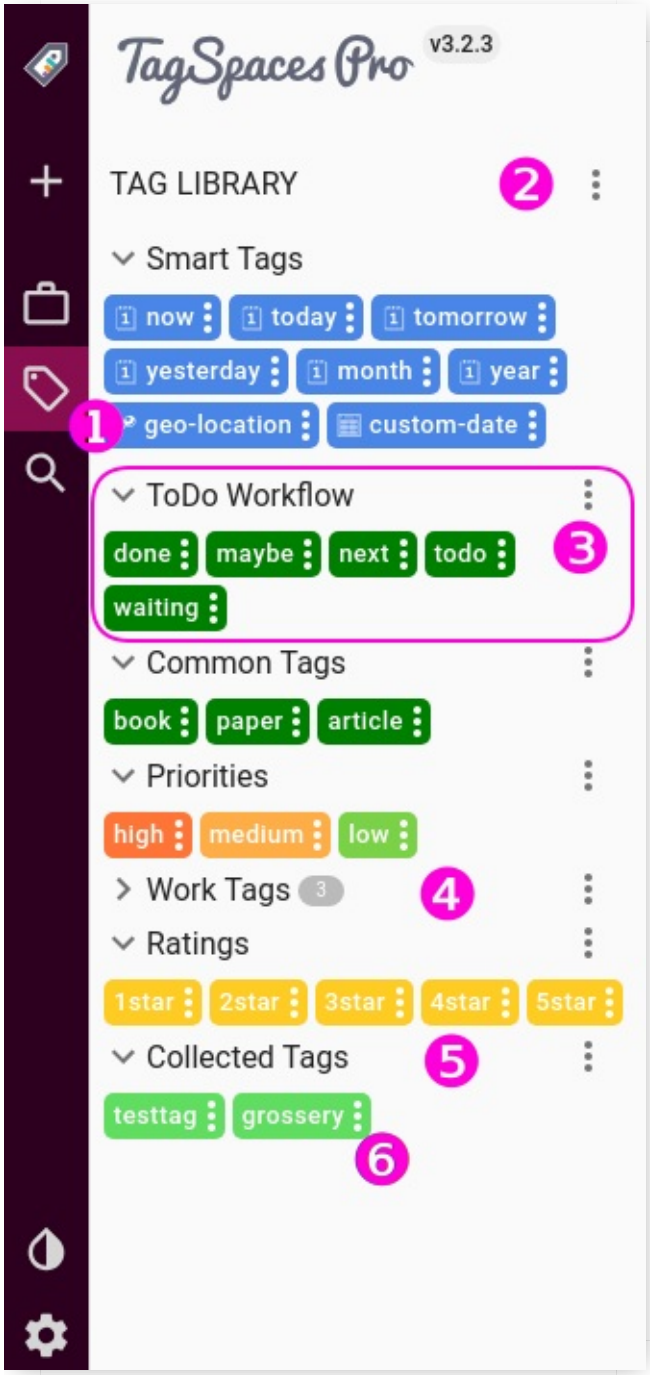
In addition to the regular locations, the cloud locations have the following properties:

- *Location Path* - the path in the bucket to which this location points
- *Access Key* - the access key of a user, who has the rights to open this bucket
- *Secret Access Key* - the secret access key of the user
- *Bucket Name* - the name of the bucket to which this location points
- *Region* - the AWS region, where the bucket is hosted. The dropdown is located on the right from the bucket name.

> Note: *Watch for external changes in this location* is disabled because it is not available for cloud locations.

Tag Library

The tag library is the place where you can manage and organize the tags with which you can tag your files and folders. To open it, you need to click the button with the tag icon (1), in the vertical navigation placed on the most left part of the application.



The tag library has its own context menu, which can be started by clicking the three dot icon on right of (2). The area marked with (3) represents a single tag group. On the top of (3) there is a three dot icon which will open the context menu of this tag group. (4) show a collapse tag group. Every tag group can be collapsed by clicking on the arrow icon in front of the tag group's name. (5) shows a special tag group used for automatically collecting new tags used in the application. (6) shows a sing tag. Every tag has its own three dot icon, which will open its context menu

Tags

The primary purpose of TagSpaces, is tagging files and organizing them. **Tags** are organic and integral parts of TagSpaces. By default, tags are stored as part of the filename, which makes them easily transferable between file and operating systems, without any need to synchronize databases, or do any extra work. A file can organically hold its own tags, as long as changing the filename is permittable.

> pro **Hint** In TagSpaces PRO, you have the option to use a hidden file (a so called sidecar file) to store tags related information. This can be useful when you are not able, or not allowed to change filenames, such as with a development project, or working collaboratively on shared documents.

Tags can be anything, form descriptive labels, to categories, ratings, priorities, or dates, timestamps, and smart tags.

> **Hint:** To learn more about the powerful tag management features TagSpaces offers, read the Tagging section

Tag groups

Tags are organized into **tag groups**, which are basically categories for tags with similar purpose and/or characteristics. Tag groups make it easier to navigate, select and manage tags in the system. TagSpaces comes with a few pre-defined tag groups, which you can freely modify, delete, or expand upon. The default tag groups are:

- **Smart tags** include various date based tags and geo location tagging. Learn more about smart tags in the tagging section
- **Common Tags**: This group contains two predefined tags, *book*, and *paper*. This can give you a general idea of the intended purpose of the group.
- **Getting Things Done**: Another useful category that can enhance your productivity, is an implementation of the GTD time management method. Predefined tags here are *done*, *next*, *maybe* and *waiting*.
- **Priorities** are exactly what they seem. These are color coded tags that will be easily identifiable by a simple glance. The three predefined priority tags are:
  - *high* - dark orange background
  - *medium* - orange background
  - *low* - lime background
- **Ratings** are predefined star-rating tags with an identifiable yellow background. As the `*` (asterix) character is considered a 'wildcard' on most file systems (meaning you cannot use it in filenames), giving a star rating as a filename-based tag would be somewhat difficult. With the yellow "rating" tags this problem can be easily solved, so your files can be organized in a rating-based system.
- **Collected Tags** is a group for all other tags you have given to your files or folders that do not belong to any other group. This tag group is created automatically after the option for collecting tags is activated in tab *General* of the application settings.

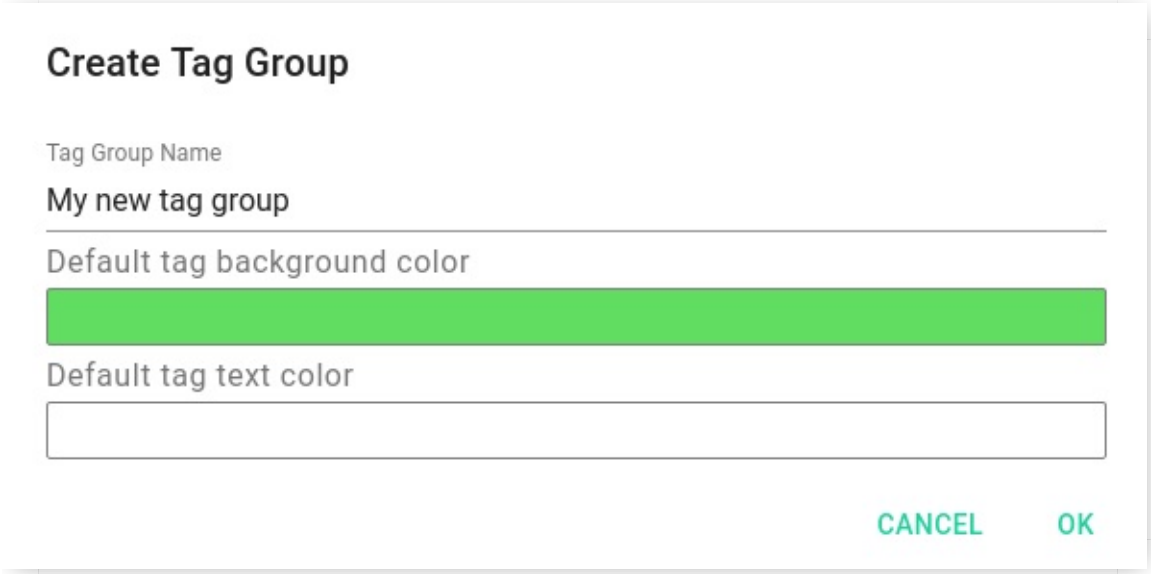
> **Hint:** All the tag groups except the smart tag are optional and can be removed from the tag library if they are note needed.

Tag library context menu

This menu contains the following three menu items:

- **Create Tag Group** - Opens the dialog for creating tag groups
- **Import Tag Library** - Start the process for importing tag groups in TagSpaces. Lear more in the tag sharing tutorial
- **Export Tag Library** - Start the process for exporting tag groups from TagSpaces. Lear more in the tag sharing tutorial

To create a new tag group, click on **Create Tag Group**, right underneath the tag groups. This will bring up a dialog, that will allow you to specify a name for your tag group, and the color of tags that belong to that group. color coding tags groups can help to easily distinguish between tags belonging to the same group, or otherwise share common characteristics. You can also specify the text color for the tags, which can be useful when choosing a lighter or darker background, so that the name can contrast.

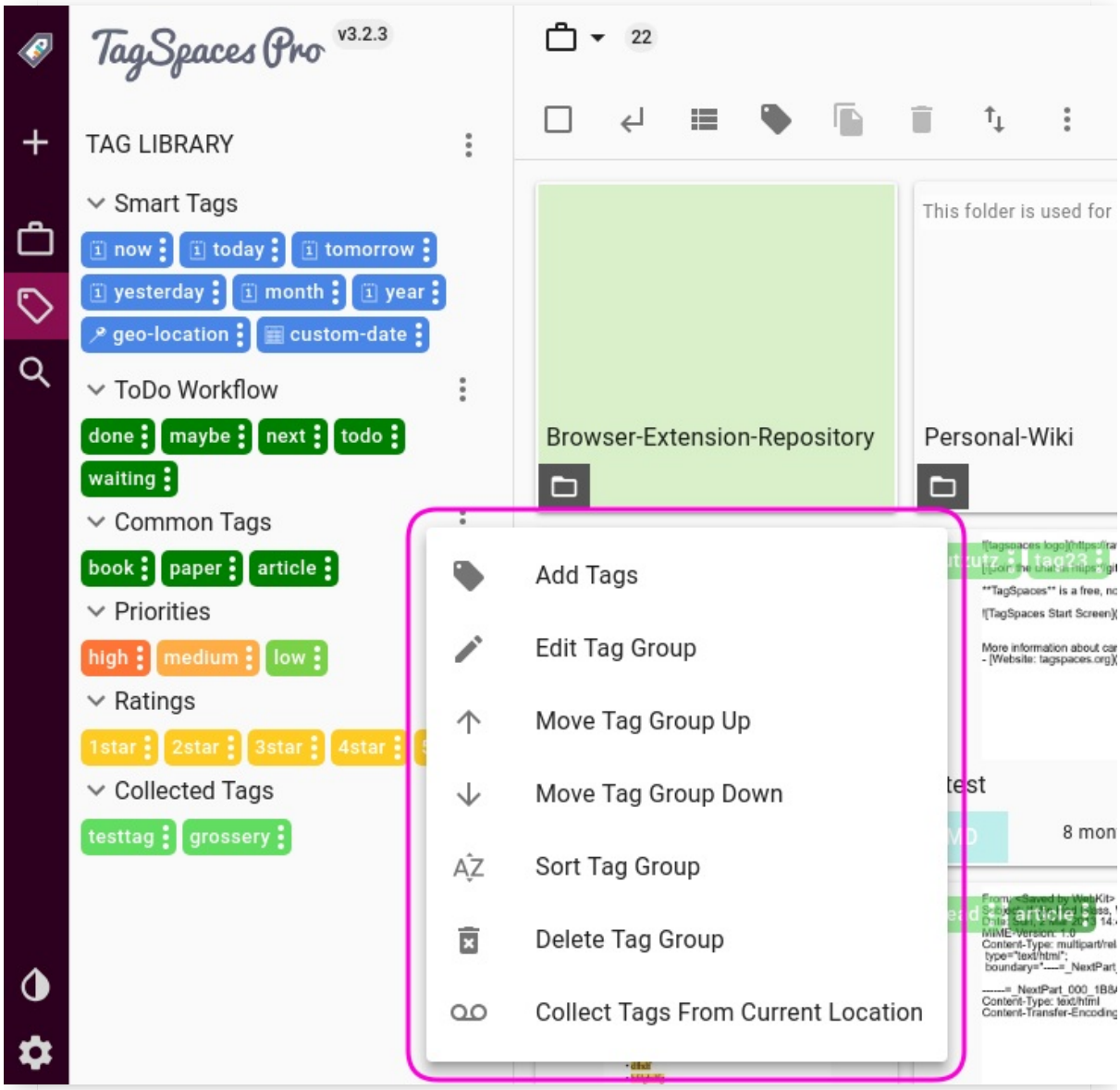


Selecting the **Import tag library** and **Export tag library** options will allow you to import/export predefined tags, which you have previously created in TagSpaces. This option takes a `JSON` file, previously exported from (probably another instance of) TagSpaces, and imports all the tags contained within, into your system. This can be useful, when you use TagSpaces on multiple systems, or you work collaboratively with others and wish to share the same tags library. To learn more about sharing tags between systems and people, refer to the Sharing tag libraries tutorial article.

Tag group context menu

Tag groups have their own context menu, accessible by clicking the three dot icon next to the tag group's name. The context menu offers the following options and operations:

- **Add tags** - This will open a very simple dialog, allowing to add new tags to the group. You can add more than one tag at once, separating them with commas, like `tag1,tag2,tag3,tag4`
- **Edit Tag Group** - will let you change the tag group's name, the default tag background, and tag text colors.
- **Move Tag Group Up** and **Move tag Group Down** - will allow you to change the tag group's position among other groups.
- **Sort Tag Group** - will sort all tags inside the group alphabetically. Useful after having manually added a number of tags.
- **Delete Tag group** - will remove the tag group from the library, with all the tags contained within.
- **Collect Tags From Current Location** pro - will collect all the tags used in the current location in the selected tag group.

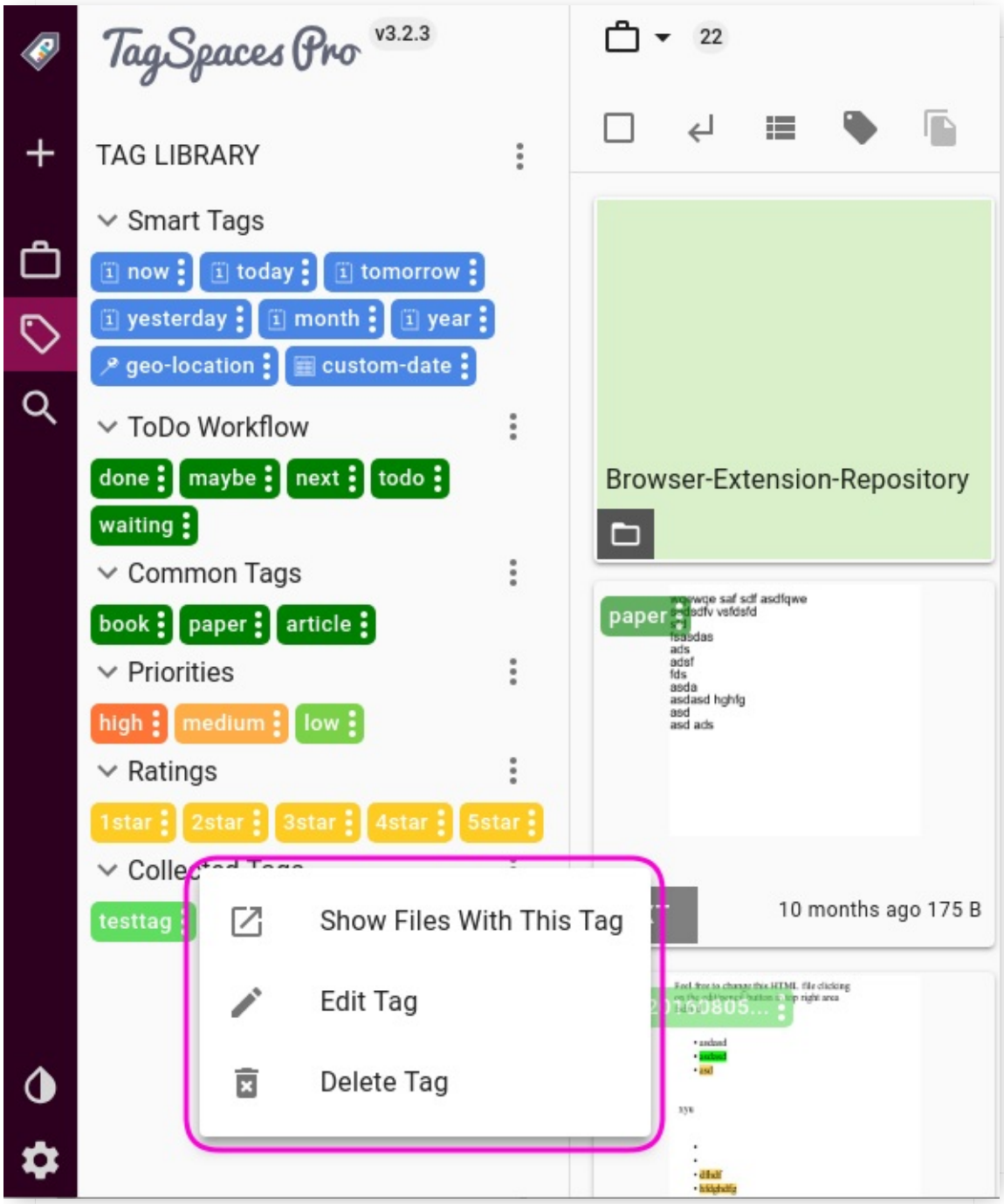


> **Hint:** Learn more about how to change the default tag background and text colors in the settings documentation section below.

Tag context menu

Each tag also offers its own context menu, accessible by clicking or right clicking on the tag. The displayed context menu will be slightly different depending on whether you've accessed it from the left panel tag library, or form a tag that is already added to a file. When accessed from the left panel, you will be presented with four options:

- **Show Files With This Tag** - is essentially a tag filter you can apply, so that only files that have the selected tag applied to them, will show.
- **Edit tag** - will let you change the tag's name, or background color and text color.
- **Delete tag** - will remove the tag from the tag group and the tag library. It will not remove it from any file or folder.



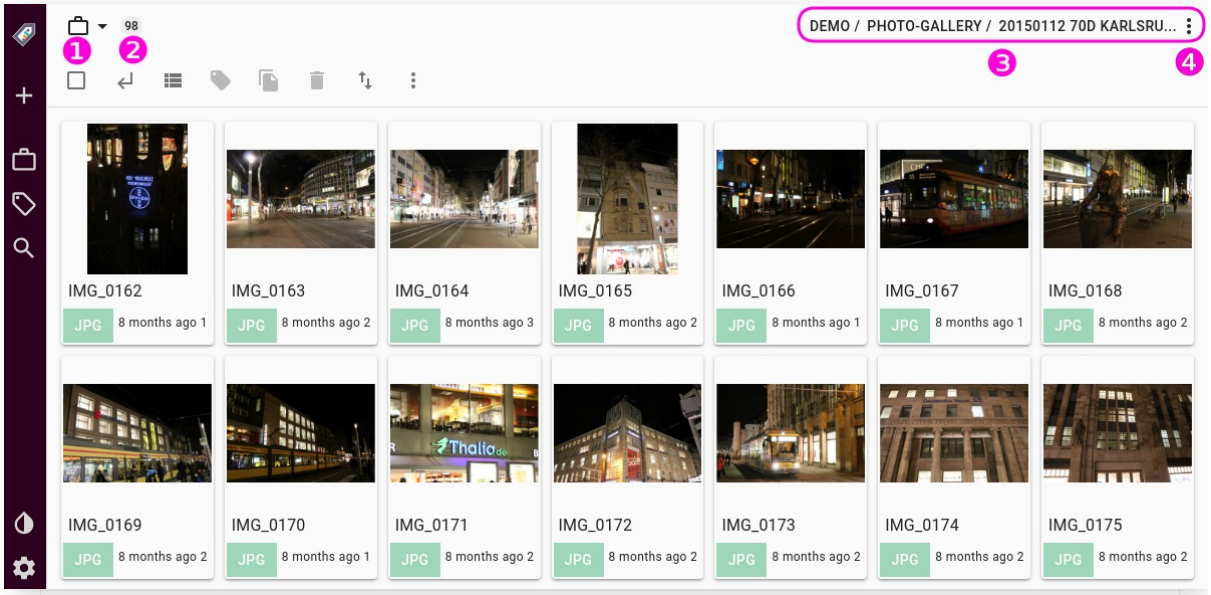
Accessing the tag context menu from the browsing area in the default perspective will show similar menu, containing the following items:

- **Show Files With This Tag** - is a tag filter you can apply, so that only files that have the selected tag applied to them, will show.
- **Edit tag** - will let you change the tag's name
- **Remove tag** - will remove the tag from the file or the folder on which the tags is assigned.

> **Hint:** Editing the name of a tag deleting one from the library tags will only affect the library itself. If the same tag had had already been applied to a file, the tag on that file will remain intact. But if you change the color of a tag in the library, it will be reflected on the files marked with that tag as well. This is because of the way TagSpaces handles tags, that essentially become an organic part of the filename. To learn more about how tagging works, please refer to File tagging based on filename under the Tagging section.

Browsing Area

The file browsing area normally takes up he majority of the user interface. This is where all files from the currently active folder are displayed, either as a list, grid, or folder tree, depending on the active **perspective**. To learn more about the all file browsing features, navigate to the File Browsing Section.

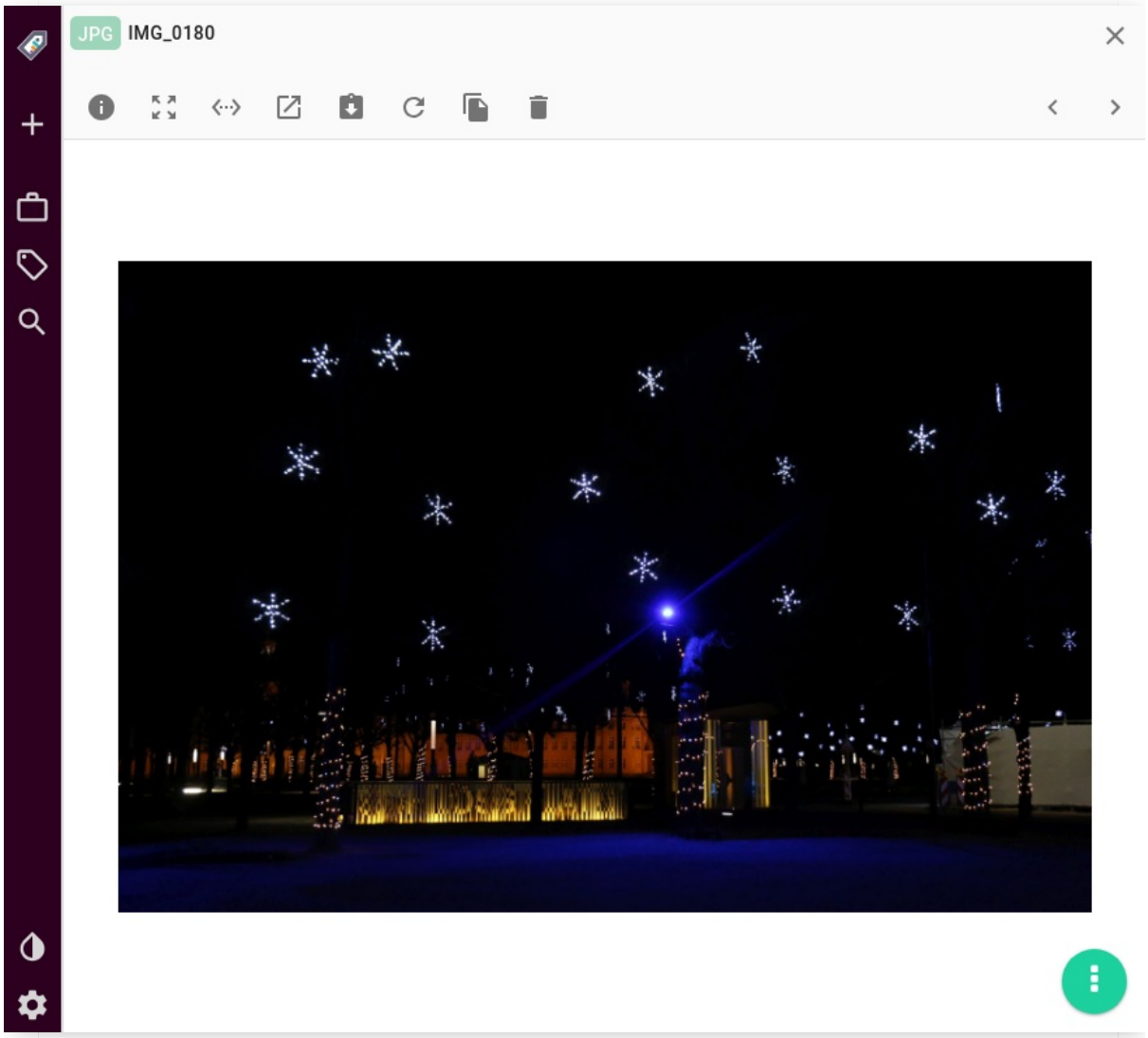


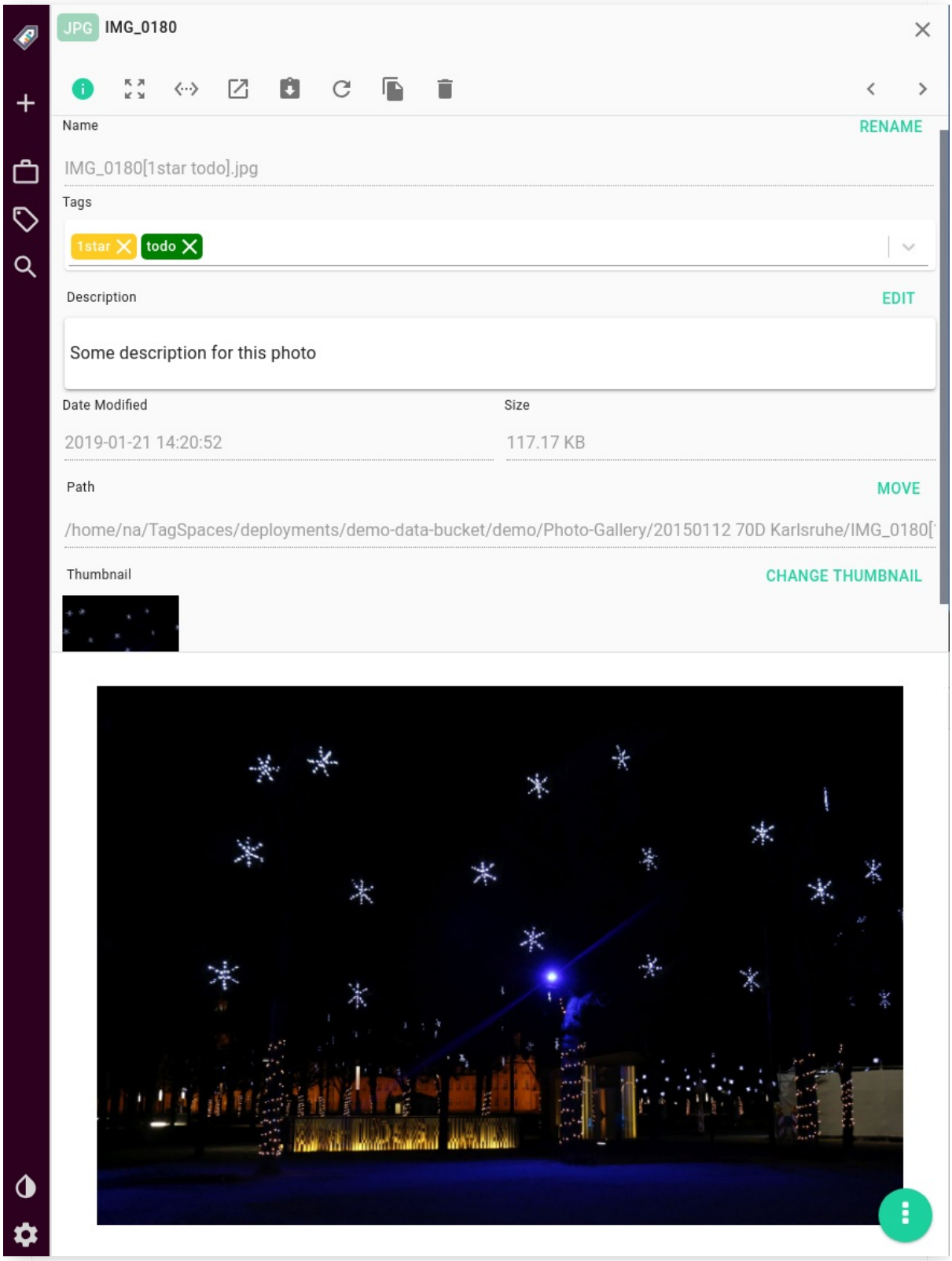
Clicking on the three dot icon above (4) will open the context menu for the current directory, which contains the following menu items:

- **Open Parent Directory** - Open the parent folder in the browsing area, only if the parent folder is located inside the current location
- **Reload Directory** - Reloads the content of the current folder
- **Rename Directory** - Opens the folder rename dialog
- **Delete Directory** - Open a dialog where you can confirm the deletion of the folder
- **Show in File Manager** - Opens the folder in the default file manager of the operating system
- **New Subdirectory** -
- **New File / Note** -
- **Add Existing File** -
- **Extract Content** -
- **Directory Properties** -

Preview and properties area

When a file is opened, the main UI area splits into two, the rightmost page becoming the **Preview area**. TagSpaces is capable of natively displaying different file types, from markdown, through HTML and docx, to PDF or various image files. To learn more about all the viewable and editable files, and TagSpaces' full file handling capacity, navigate to the Viewing Files section and Editing Files section respectively.

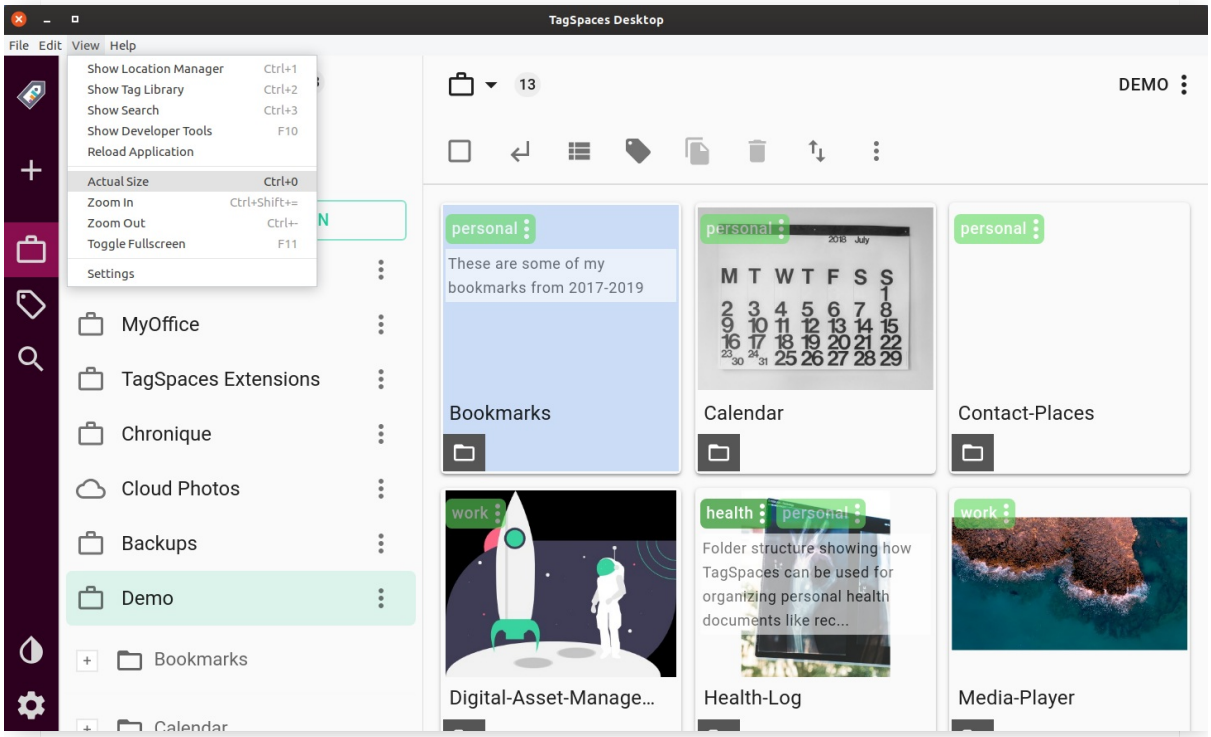

JPG
IMG_0180



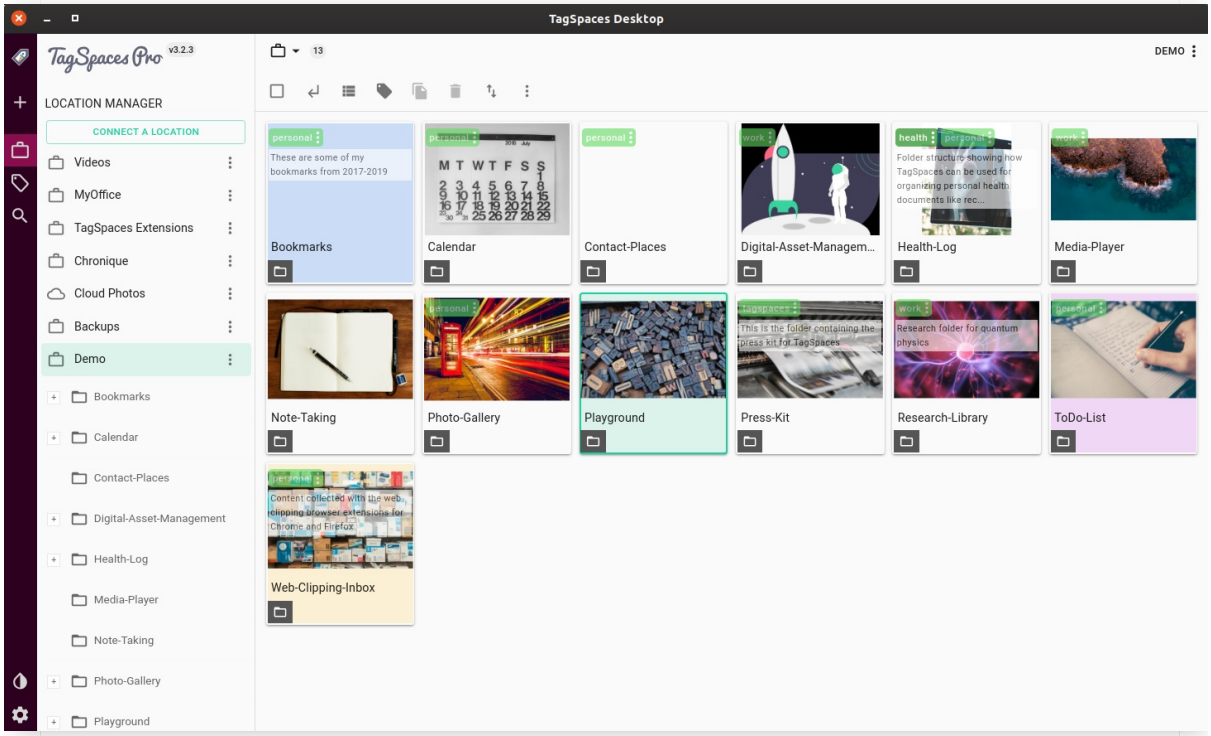
Zoomable user interface

The user interface of the application zoomable, this allows you to adjust the zoom scale according to your screen resolution and display size. In the desktop you can access the zoom functionality from the **View** in the **Main Menu**. Here you will find zoom in, zoom out functionality and also the possibility to restore the initial zoom factor with **Actual size** menu item.

In the following screenshot you will see the menu options from the **View** sub menu and also TagSpaces with zoomed in interface.



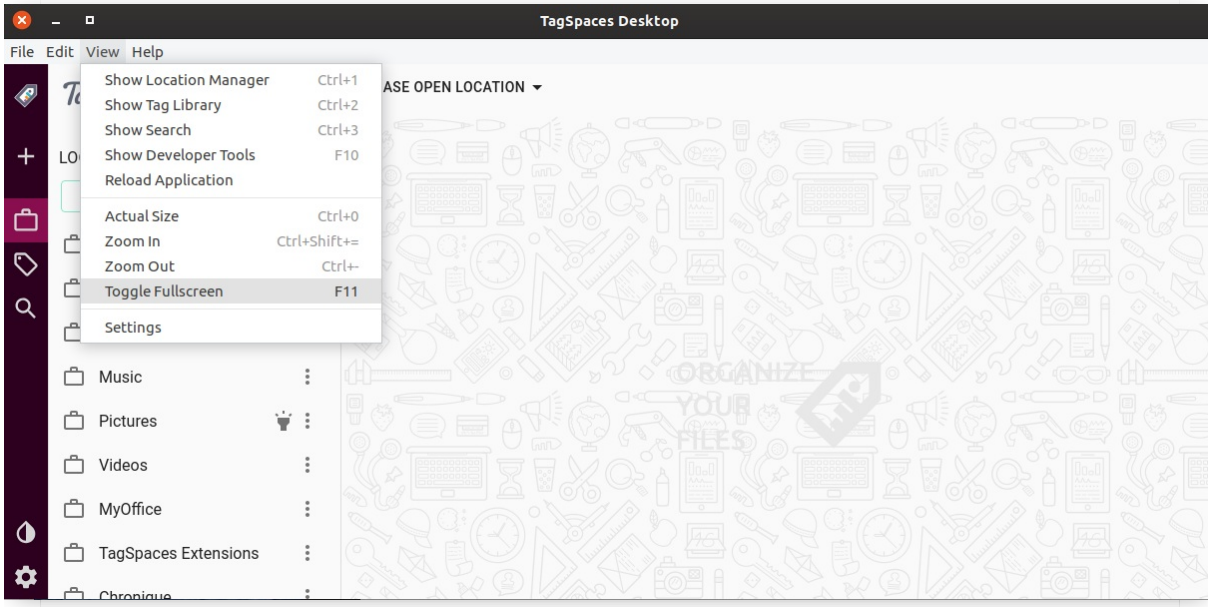
And here you will see TagSpaces with zoomed out interface.



> **Hint**: If the application is running in a browser, please use the integrated in the browser zooming capabilities.

Full screen mode

The desktop application can run in a full screen mode, which can be toggled by pressing the `F11` key, or selecting *View -> Toggle Fullscreen* from the **Main Menu.**



> **Hint**: If the application is running in a browser, please use the integrated in the browser full screen mode capabilities.

Keybindings in TagSpaces

Keyboard shortcuts are combinations of two or more keys that you can use to perform a specific task that would typically require more than one mouse click. They are very useful and can make it easier to work with your software, saving you time and effort as you work with TagSpaces and other applications.

To use a keyboard shortcut, hold down one or more keys while pressing the last key of the shortcut. For example, to use the shortcut **Ctrl** + **C** on Windows and Linux or **Command** - **C** on macOS, hold down **Ctrl** button, press **C** and then release both keys.

TagSpaces like a much of modern software also provides a range of keyboard shortcuts for better experience with it. Please find the list of the currently supported keyboard shortcuts bellow.

Default keybindings

General

Action	Key binding on Windows/Linux	Key binding on macOS
Show Keybinding Help	`F1`	`F1`
Fullscreen mode	`F11`	`F11`
Show developer tools	`F10`	`F10`
Show location manager	`control + 1`	`⌘ + 1`
Show tag library	`control + 2`	`⌘ + 2`
Show search panel	`control + 3`	`⌘ + 3`
Show search panel (alternative)	`control + f`	`⌘ + f`

Working with documents

Action	Key binding on Windows/Linux	Key binding on macOS
Open current file or folder	`enter`	`enter`
Rename current file or folder	`F2`	`F2`
Delete current file(s) or folder(s)	`del`	`?`
Tag current file(s) or folder(s)	`control + t`	`⌘ + t`
Select/open the next document	`right` or `down`	`right` or `down`
Select/open the previous document	`left` or `up`	`left` or `up`
Select/Deselect all files	`control + a`	`⌘ + a`
Edit the currently opened document	`control + e`	`⌘ + e`
Save the currently opened document	`control + s`	`⌘ + s`
Close the currently opened document	`control + w`	`⌘ + w`
Open the properties of the currently opened document	`alt+enter`	`alt+enter`

Global keybindings

These keybindings are available even if TagSpaces is currently not in focus, but running in background for example in the system's tray.

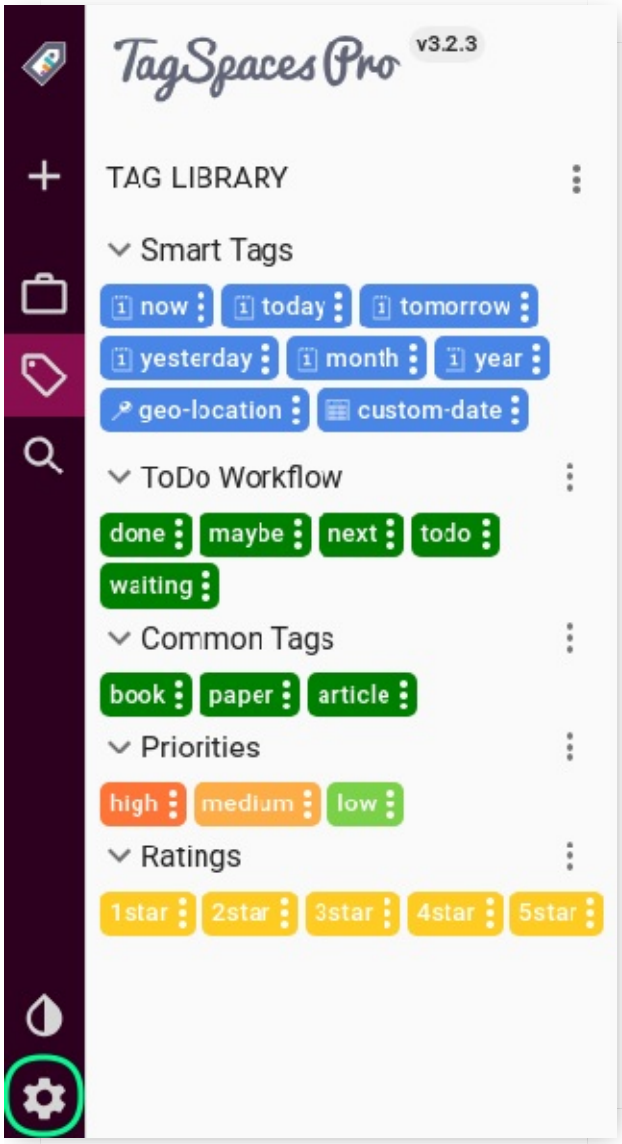
Action	Key binding on Windows/Linux	Key binding on macOS
Show TagSpaces	`control + alt + w`	`⌘ + alt + w`
Create new text document	`control + alt + n`	`⌘ + alt + n`
Open next document	`control + alt + d`	`⌘ + alt + d`
Open previous document	`control + alt + a`	`⌘ + alt + a`
Pause/resume playback	`control + alt + p`	`⌘ + alt + p`

Configuring the keybindings

Most of the keybindings integrated in TagSpaces are configurable. To change them, open the **Settings**. In the settings dialog, open the tab **KEY BINDINGS**. Learn more in the settings part of the documentation.

Settings

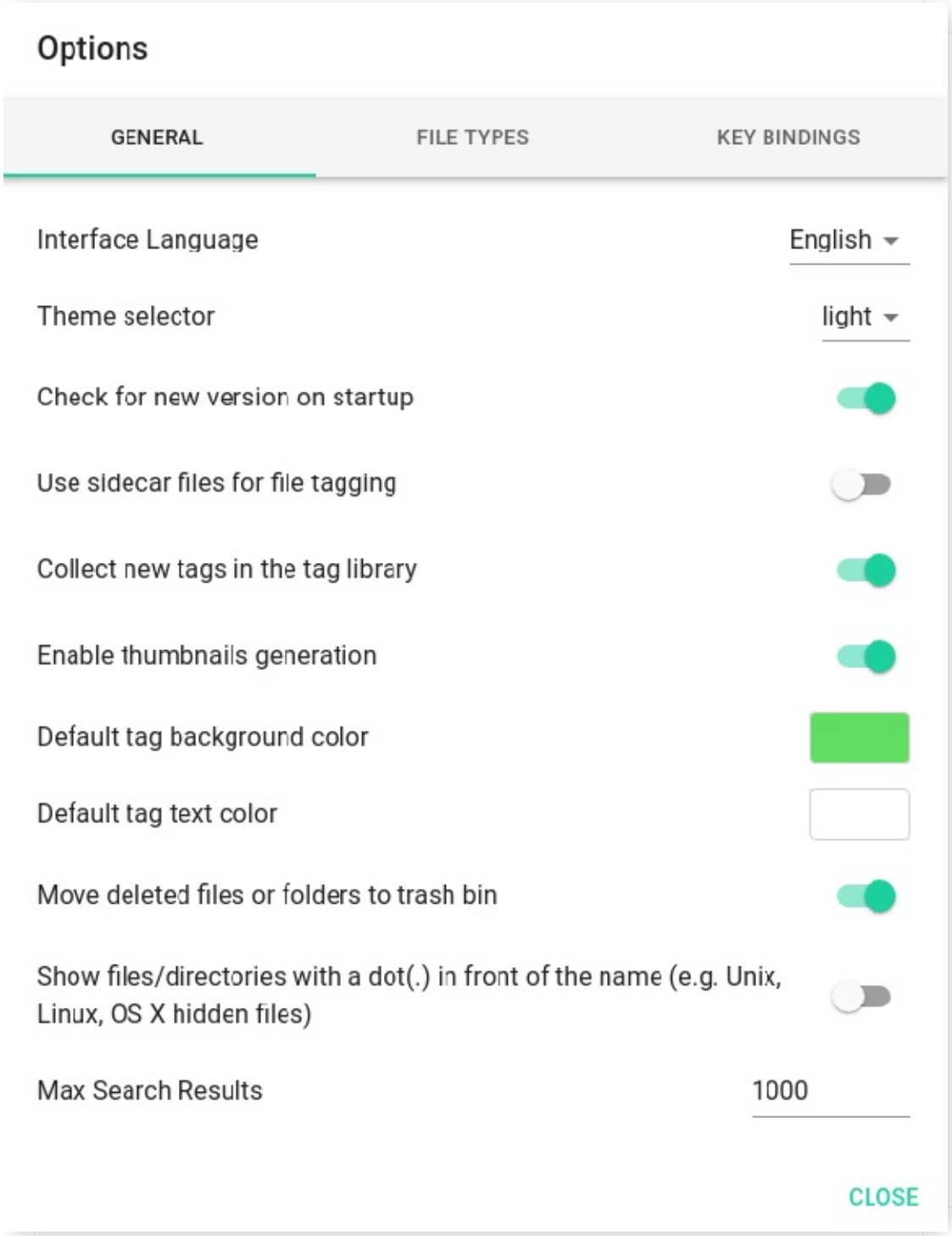
To access the settings dialogue, just click on the gears icon at the bottom left part of the app.



The dialogue has three tabs, which will be described in order:

General

Most of the TagSpaces' common settings can be accessed from this tab.

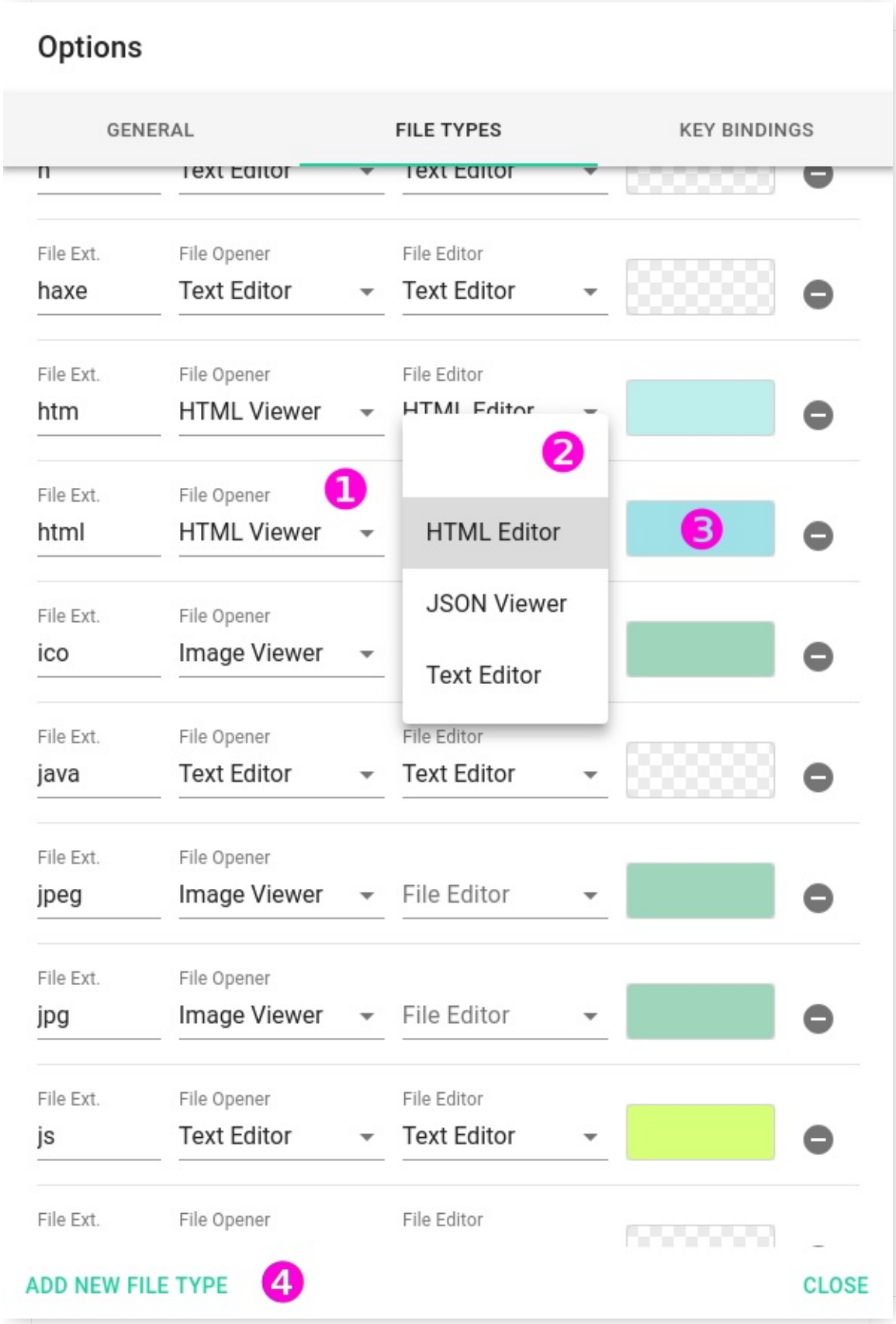


- **Interface language** - TagSpaces has been translated to a variety languages. Choose your preference here
- **Theme selector** - Here the user interface theme can be selected.
- **Check for new version on startup** - You will be notified if a new version if available
- **Use sidecar files for tagging** - once activated, tags assigned to files will not be embedded in their file names, but in so called sidecar files, located in hidden .ts sub folder of the folder where the current file is located.
- **Enable thumbnails generation** - will generate and store persistent thumbnails in the hidden .ts sub folder of the folder where current file is located.
- **Default tag background color** - Allows you to change to default background color of any newly created tags. Existing tags will not be affected.

- **Default tag text color**- Allows you to change to default text color of any newly created tags. Existing tags will not be affected.
- **Move deleted files or folders to trash bin** - will allow you to recover deleted files later, from the trash/recycle bin of your operating system.
- **Show files/directories with a dot(.) in front of the name** - This will allow the browsing of hidden files in UNIX-like systems (Linux, macOS, BSD, etc.)
- **Max search results** - This is the limit of the maximum files and folders, which can be shown in the perspectives and respectively the maximum count of search results.

File types

Here you can choose which viewer and editor to launch for each supported file type.



When you click on a file type's viewer dropdown, you will be offered a list of all installed viewer plugins (**1**).

> **Note:** To learn more about file previews, refer to the **Viewing Files** section.

When you click on a file type's editor dropdown, you will be offered a list of all installed editor plugins (**2**).

> **Note:** To learn more about file previews, refer to the **Editing Files** section.

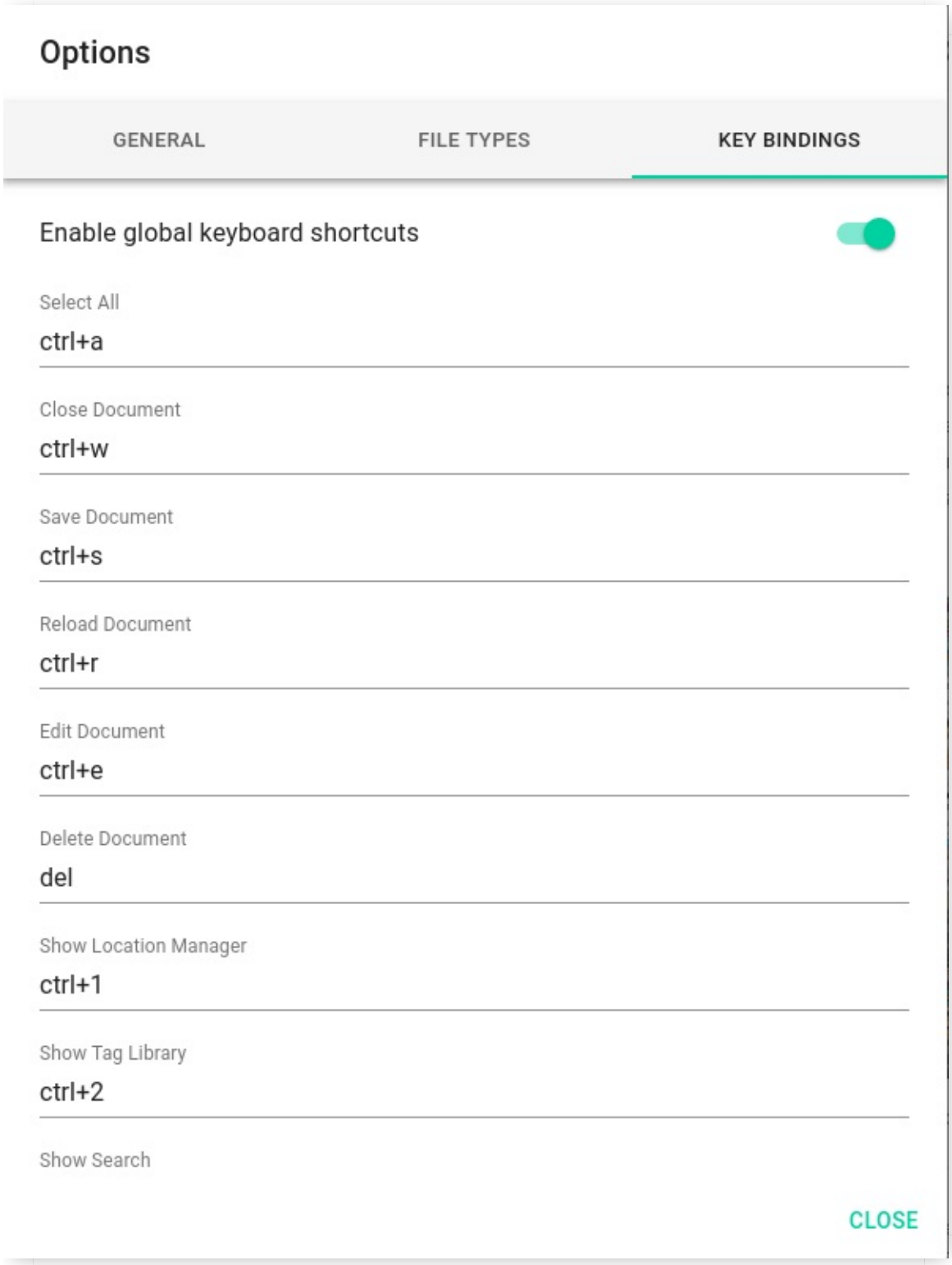
Clicking on the colored area (**3**) in the most right part of every files type will open a color selection dialog where you can adjust the color of the current file type.

To add a formerly unrecognized extension, just use the **Add New File Type** button (**4**).

Key bindings

Under the key bindings tab, you can view and edit all the available key bindings, excluding those assigned to tags.

Most of the keybindings are configurable. To change them, Open the **Settings**. In the setting dialog, open the tab **KEY BINDINGS**.



Here you can see all the keyboard shortcuts by default in TagSpaces, and what actions they perform. You also have the ability to change keyboard shortcuts by clicking on the field in which they were written. Before changing any of the binding, please read the following short instructions:

- For modifier keys you can use `shift`, `ctrl`, `alt`, `option`, `meta`, `command` and `mod`. The last one is special modifier allowing us to refer to the control and command key on Windows/Linux and macOS operating systems respectively.
- Other special keys are `backspace`, `tab`, `enter`, `return`, `capslock`, `esc`, `escape`, `space`, `pageup`,

`pagedown`, `end`, `home`, `left`, `up`, `right`, `down`, `ins` and `del`.

- Any other key you should be able to reference by name like `a`, `/`, `$`, `*`, `=`.
- You can assign many keys to a single operation, simply by separating them by `,`.

> **Note:** To learn more about the key bindings, refer to the this section of the documentation.

Internationalization of the application

The user interface of TagSpaces application was translated so for in 26 languages by volunteers. For the coordination of the translation process we utilize the great service of Transifex. Since TagSpaces is evolving fast, we need support for the translations of the existing languages and of course we are happy to see when new languages are added. If you want to support the translation efforts please create a Transifex account and apply for membership in the TagSpaces project there. The following table lists the languages in which the application is already translated.

ISO	Language	Language in English
en_US	English	English
de_DE	Deutsch	German
it	Italiano	Italian
zh_CN	中国的	Chinese
bg	Български	Bulgarian
ja	日本の	Japanese
pt_BR	Português	Brazil
pt_PT	Português	Portugal
fr	Français	French
sk_SK	Slovenský	Slovak
es	Español	Spanish
uk	Український	Ukrainian
ru	Русский	Russian
tr	Türk	Turkish
cs	Čeština	Czech
ca	Catalan	Català
ko	한국의	Korean
el	ελληνικά	Greek
sv	svenska	Swedish
nl_NL	Nederlands	Dutch
zh_TW	台灣	Chinese Taiwan BIG5
hu	Magyar	Hungarian
id_ID	bahasa Indonesia	Indonesian
pl	Polski	Polish
mt	Maltese	Maltese
hy	հայերեն	Armenian

The order of the languages represents the order in which the translations were made by the contributors.

Browsing Files and Folders

When you navigate to a folder in your active location, the files contained in the selected folder will be displayed on the main file browsing area of the user interface. TagSpaces offers flexible views to display your files. These views are called **perspectives**.

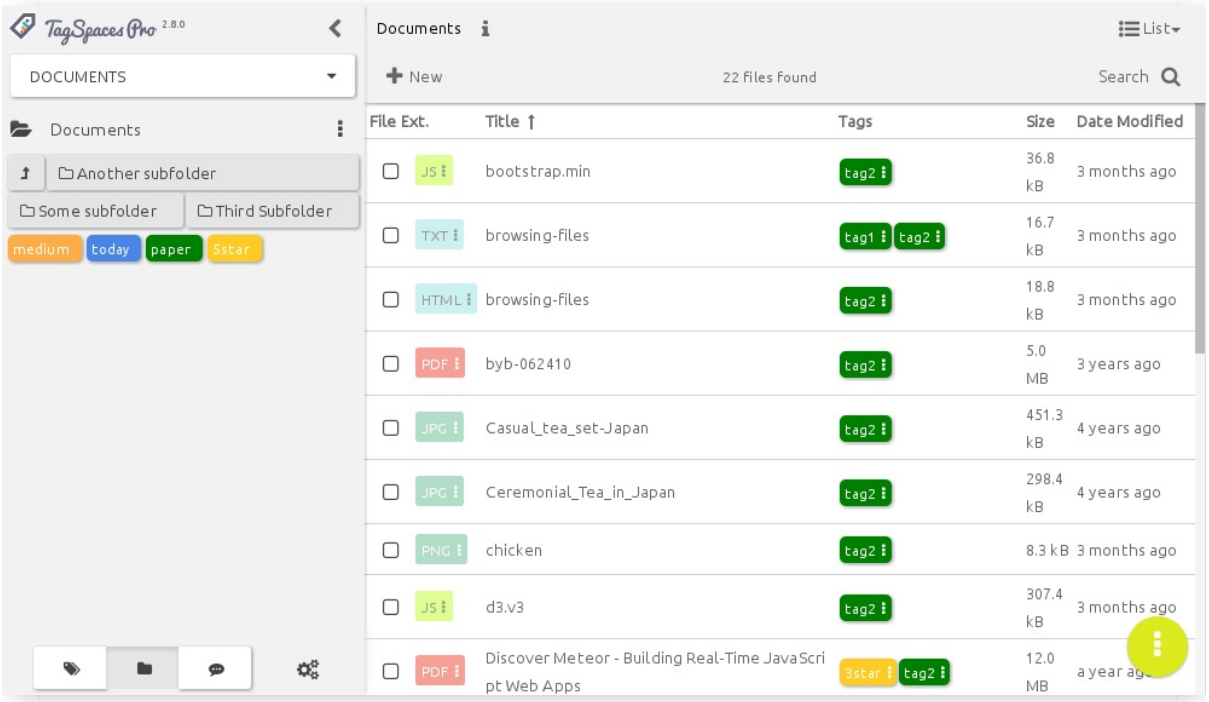
Perspectives overview

Perspectives are not an integral part of TagSpaces, but exist as modular extensions. This modular approach allows for more flexibility, easier development, and customizability of each separate perspective. TagSpaces is delivered by default with one perspective. Other perspective can be added on demand in the Enterprise version of the product. In this document we will describe the following perspectives:

- Default Perspective - Presenting your files as list and in grid.
- FolderViz Perspective (Discontinued) - This is an experimental perspective which applies some information visualization concepts to presenting your folder and file structures.

Default perspective

The most common way of presenting files in a folder is in a list and grid formats, both formats are supported in the default perspective.



Each row of the list consists of columns, which give specific information about the files. The column names can be found in the header row, at the top of the list.

The columns, from left to right are:

- **File extension** - A color coded icon, representing the file type. Scroll down to Common features -> Color coded file extensions to learn more about this feature.

  The checkbox underneath the file extension icon can be used to easily select one or more files.

  > **Hint:** This also means that `Ctrl or Shift + click` will **not** work for selecting multiple files in the current version of TagSpaces, although this functionality is planned for a future release.

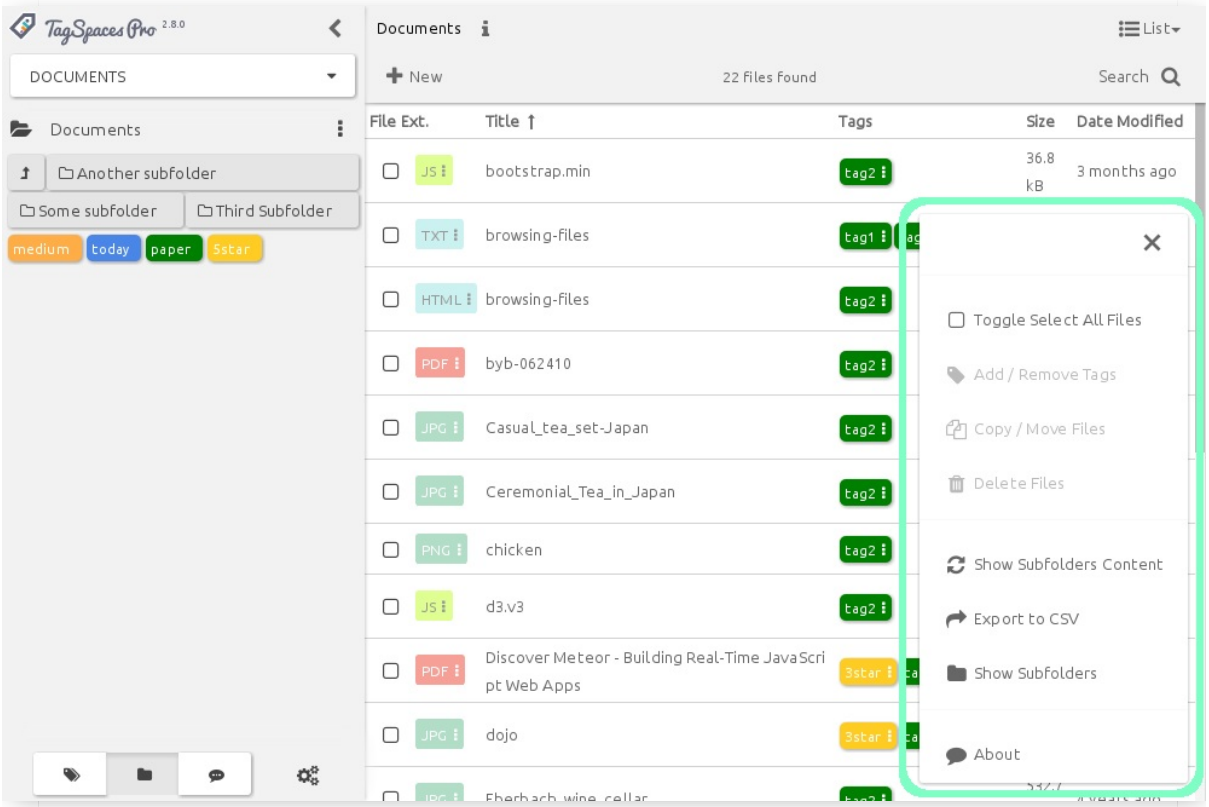
- **Title** - The file's title is the filename without the extension or any tag information. By default, TagSpaces stores tags as part of the filename. To learn more about how this is achieved, see the Tagging section.
- **Tags** - All the tags that are applied to the file will appear here, with the right background and font color. To learn more about tag colors, refer to the User interface -> Tag Library section.
- **Size** - This is the file size in a human readable format.
- **Date modified** - The time the file had been last modified, in a format of `YYYY.MM.DD-hh:mm:ss`

To change the list order, you can click the list column headers. For example if you want to sort the list alphabetically by name *(A-Z)*, click on the name. To change the direction of the sorting *(Z-A)*, click on the name again. Each header offers bidirectional sorting based on its column's properties.

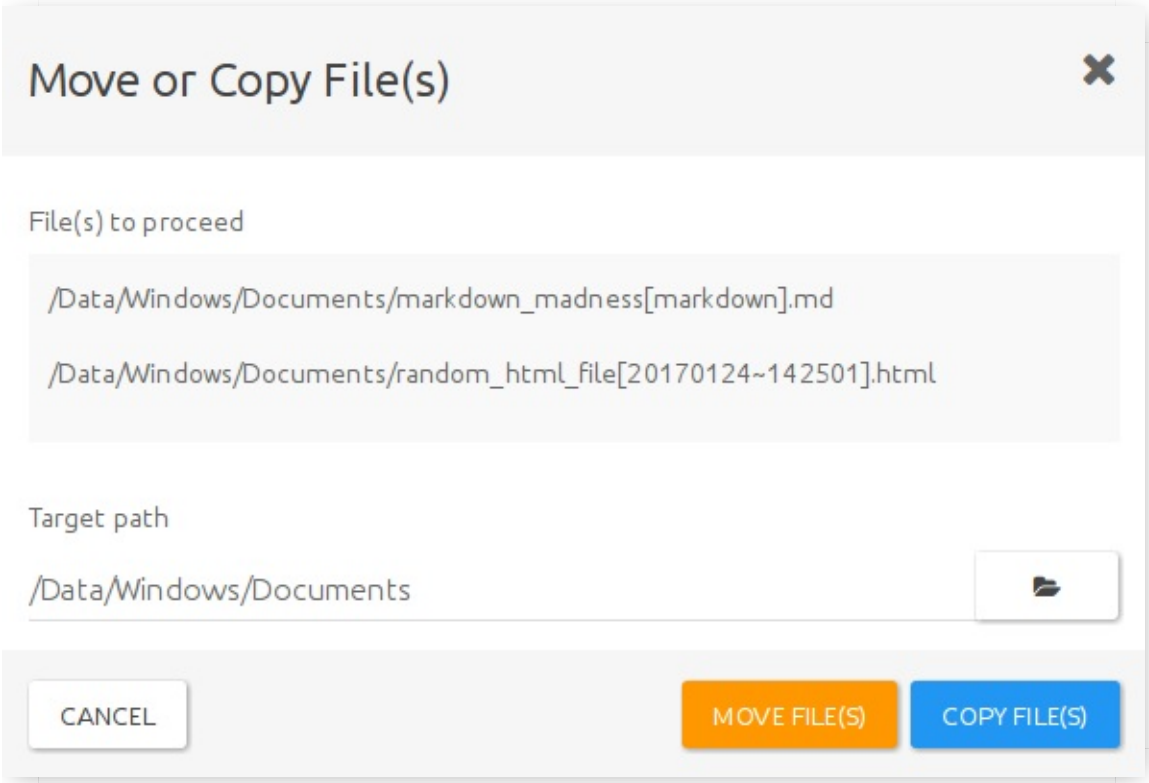
> **Hint:** A small arrow next to the name of the appropriate column, will indicate the direction of sorting, and the column to sort by.

List overflow menu options

The **Floating Action Button** (FAB) at the bottom right corner, hides a contextual overflow menu applicable for the currently active perspective. The List perspective offers its own set of options, that will allow you to manage the list of files in certain ways.



- **Toggle Select All Files** will mark every file on the list as selected or unselected.
- **Add/Remove tags** will allow you to manage tags on selected files. (This options is only available when there is an active selection, and works on multiple files simultaneously.) To learn more about how tagging works, refer to the Tagging -> Tagging using context menus section.
- **Copy/move files** will allow you to copy or move one or more selected files. (This option is only available when there is an active selection, and works on multiple files simultaneously.) Selecting this option will present you with the **Move or Copy File(s)** dialog. After specifying the target directory (which can be anywhere on the file system, even outside your connected locations), you can choose to move or copy the file(s) by pressing the corresponding button.



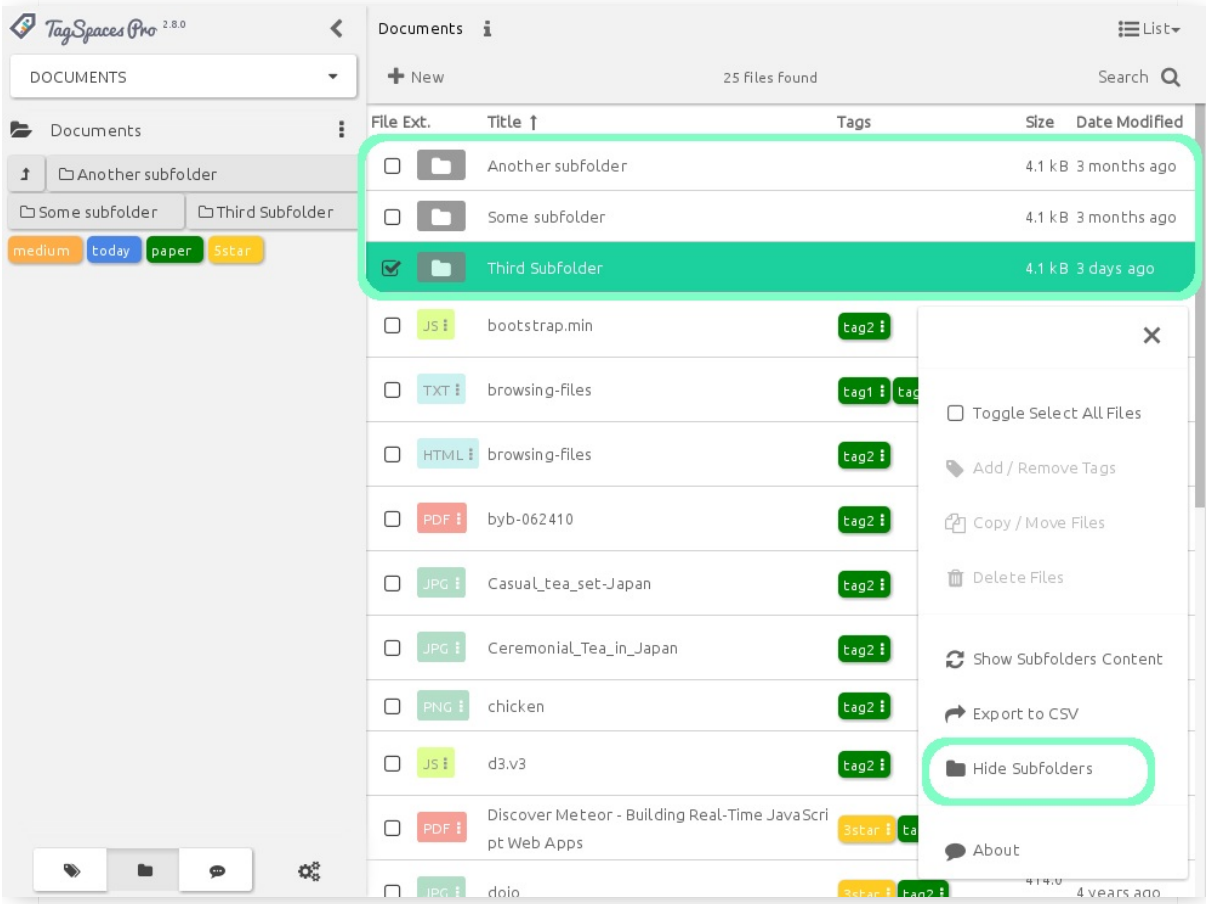
- **Delete files** will allow you to delete the file(s) selected. (This option is only available when there is an active selection, and works on multiple files simultaneously).

  > pro **Hint**: TagSpaces PRO offers an option to move deleted items to trash bin instead of deleting them permanently. To learn more about how to activate this feature, refer to the Settings -> PRO features section.

- **Show subfolders content** - If you don't care about folder structure, or do not know where in the hierarchy you would find a necessary file, you can turn this option on, to show every file from all subfolders recursively, starting from your current directory. The list will display all files in order, without any indication of their actual locations. This option will allow you to work with multiple files across a folder hierarchy at once.
- **Export to CSV** will create a **Comma Separated Variable** from all displayed files, be it from the current folder, or from the full folder hierarchy, and save it n a folder of your choice. The generated CSV contains `path,title,size,tag1,tag2,tag3,...`, where each new line represents a separate file.
- **Show/Hide Subfolders** Starting with **TagSpaces 2.8**, it is possible to list and navigate subfolders in-line with the file list.
- **About** will display information about the perspective extension in a pop-up window.

Navigating folders from from the file list

When you enable the option to *Show Subfolders* from the overflow menu, the list perspective will show all subfolders in the current folder, in-line with the file, on the top of the list.

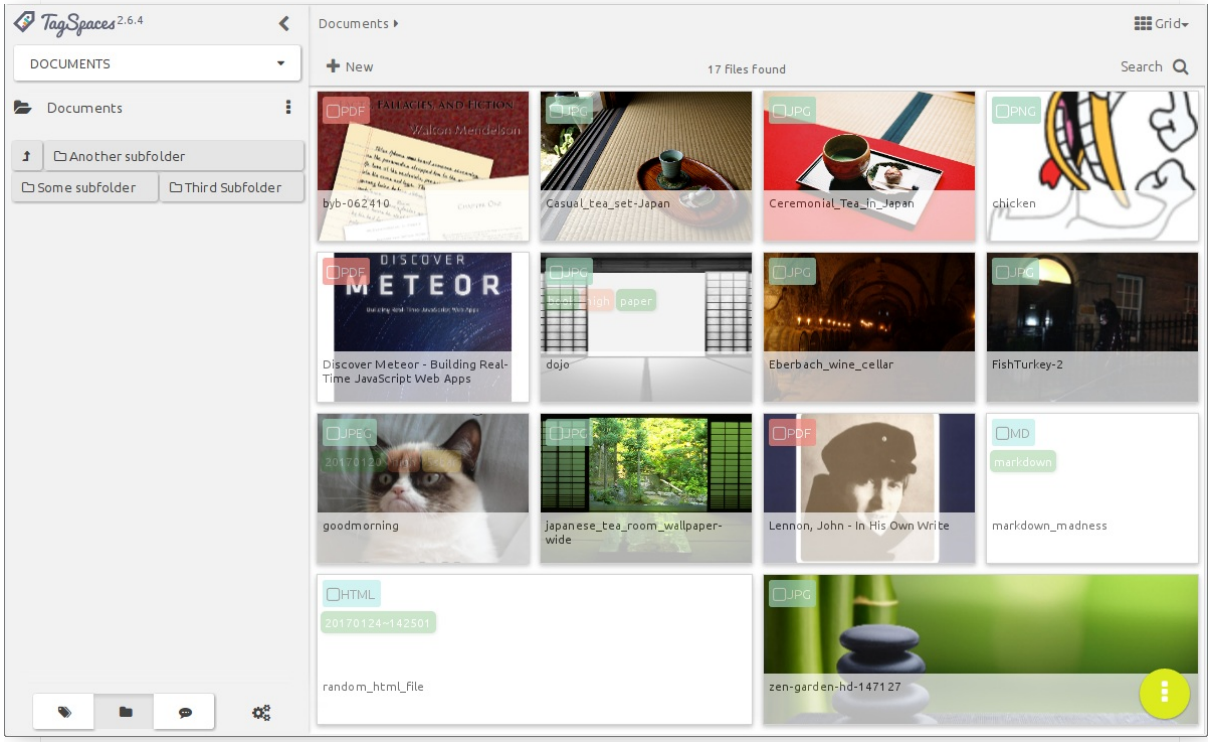


Double-clicking a folder name will change to that folder, while single-clicking it will select the folder, the same as selecting a file.

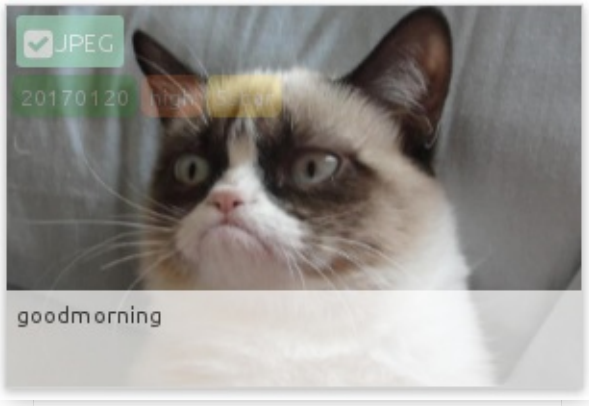
> **Note:** The right-click menu options for folders are currently the same as they are for files

Grid view

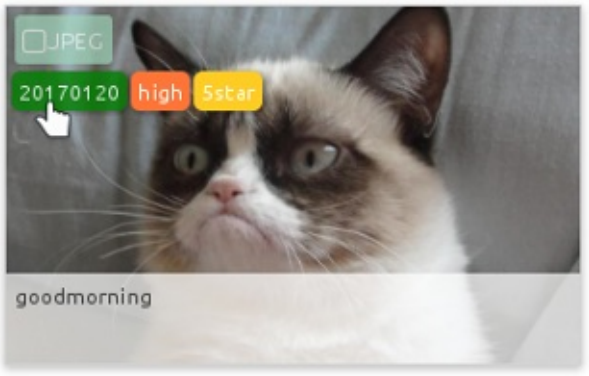
An alternative arrangement, commonly found in file browsing applications is the grid. In TagSpaces the **Grid Perspective** offers a resizeable grid with thumbnail previews of certain file formats, for quick and effective browsing.



Each card on the grid has three main components. The main body of the card shows a thumbnail preview of the file (if available), the bottom area displays the filename, the color coded icon in the top left corner represents the file extension, with a checkbox for selecting the file. Underneath the file extension icon, the applicable tags are displayed. In this view, there is no information given about file sizes, or last modified dates.

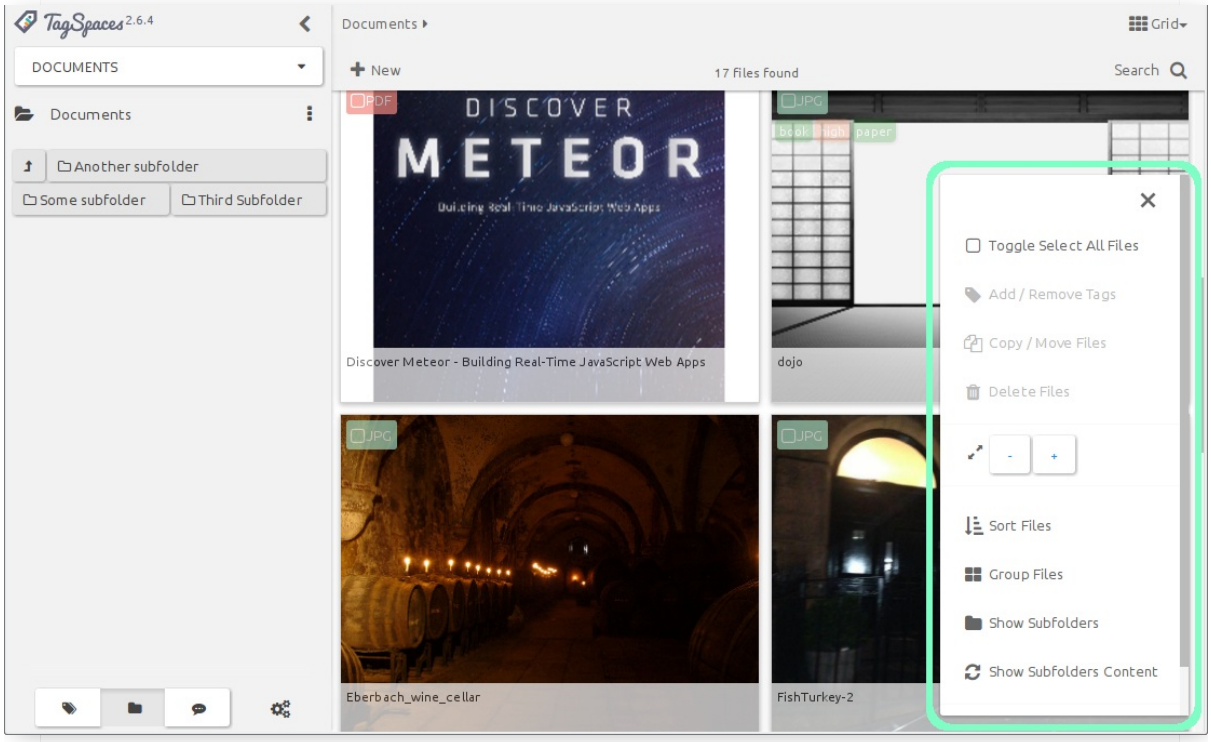


The file extension, and the tags are dimmed on the cards by default, which might render them unreadable, when there is a file preview available. To improve readability, hover your mouse over the extension, or the tags to make them more opaque.



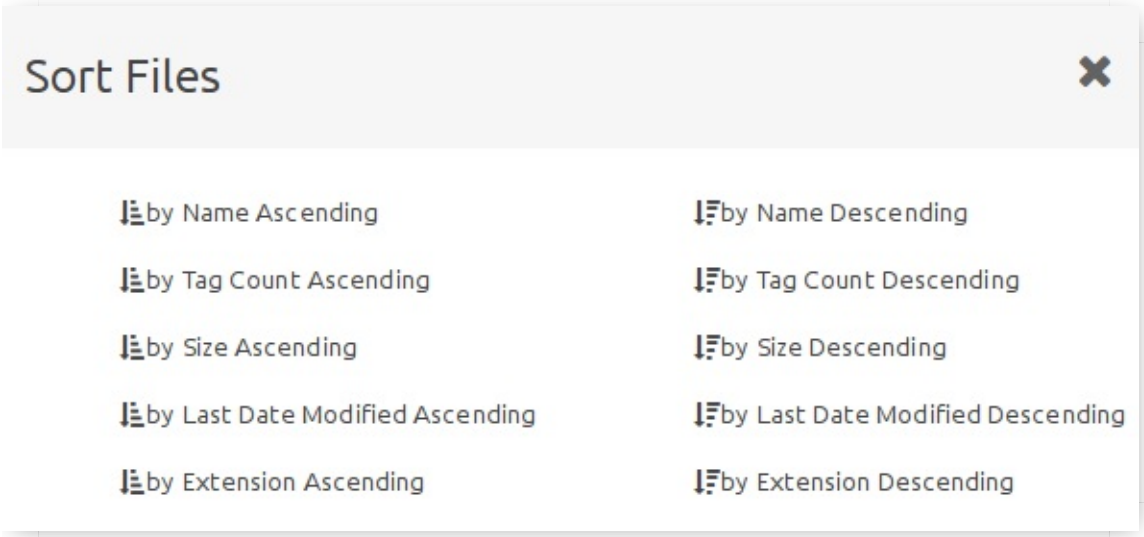
Grid overflow menu options

Just like the List perspective, the grid also offers a specific set of options, found in the overflow menu, which can be accessed by pressing the FAB.

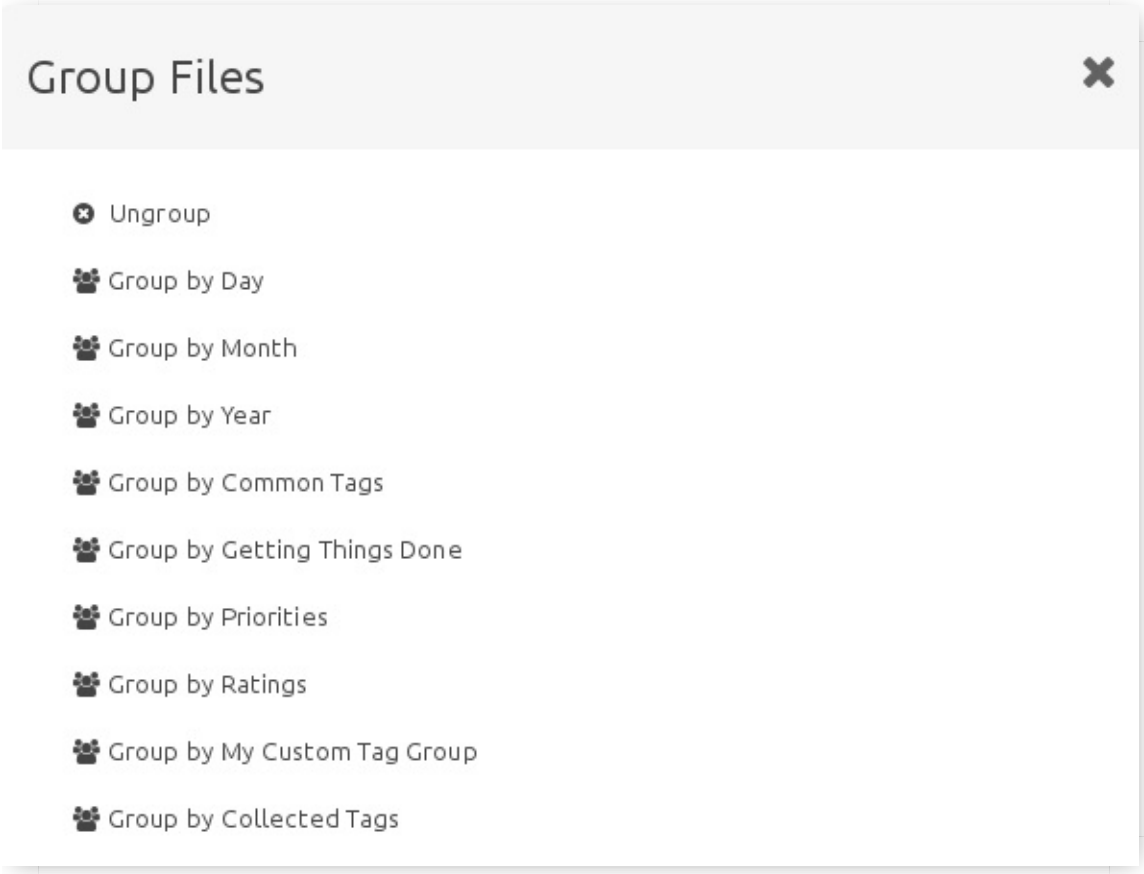


Some options (such as *Add/Remove Tags*, *Copy/Move Files*, *Delete Files*, *Show Subfolder Content*, and *About*) are identical to those of the List Perspective. The **Grid Perspective** specific specific options, which allow you to interact with the view are:

- **Change card size** - by pressing the *plus*  and *minus*  buttons, you can change the size of the cards displayed on the grid.
- **Sort Files** will present you with different options to sort files by.



- **Group files** allows for grouping based on time (either stamp tags, or last modified date accessed from the file itself), or by any defined tag groups. This will allow tag groups to also serve as a basis for file grouping, besides grouping tags themselves. (To learn more about tag groups, refer to User Interface -> Tag Library section.

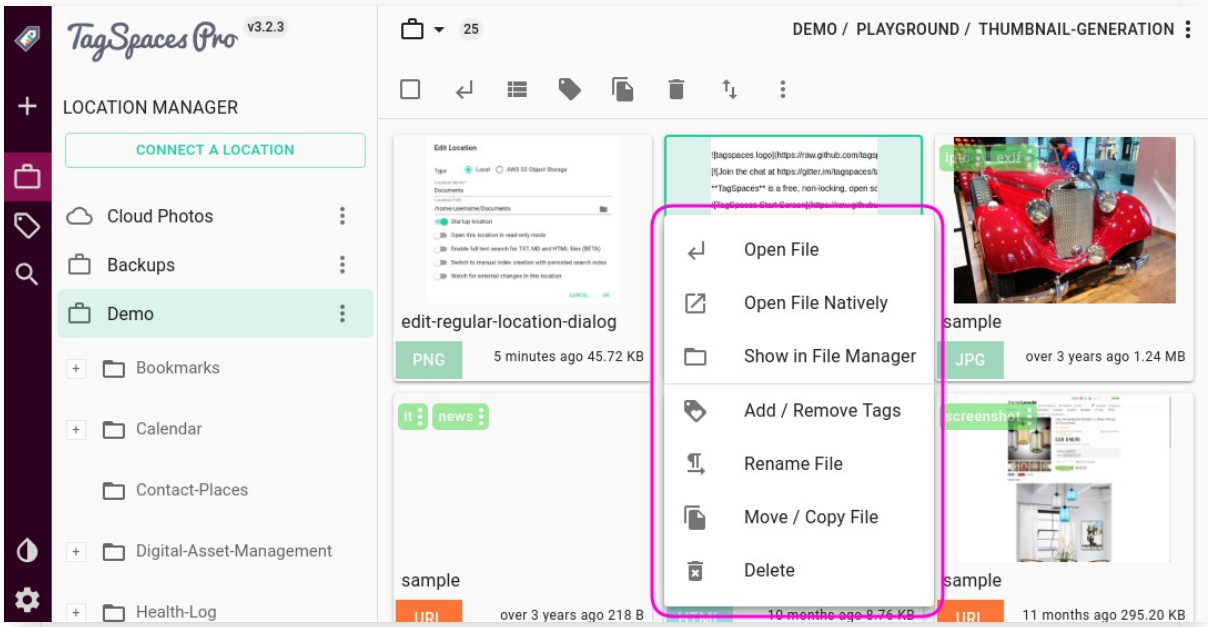


Common features

Besides the very specific functionality and features List and Grid perspectives offer, there are some common characteristics that are uniform across the two.

File context menu

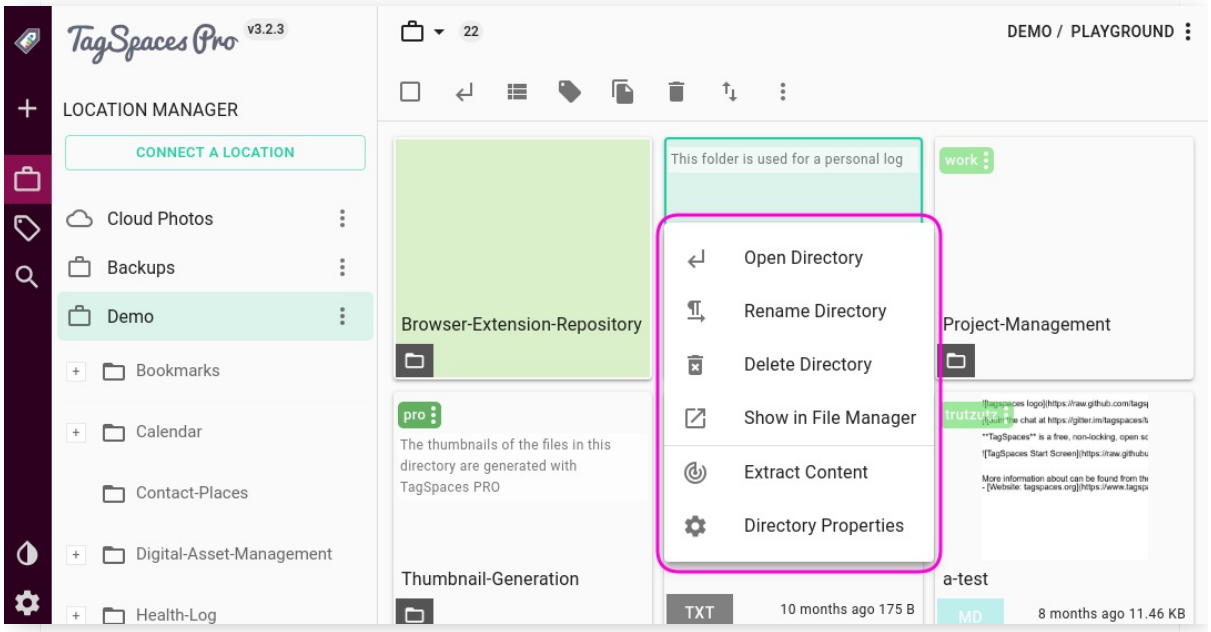
The file context menu can be accessed by right-clicking a file in either perspective. It will offer some common file management options.



- **Open file** will split the main area of TagSpaces into two, and open the file in the right pane. TagSpaces offers viewers for various file types, which can be opened inside the application. To learn more bout supported types, refer to the Viewing Files section.
- **Open File Natively** will open the file in whatever application is associated with the file type in your operating system.
- **Show in File Manager** will show this file in the default file manager of your operating system
- **Add / Remove Tags** will open a dialog where you can add or remove multiple tag to this file.
- **Rename File** will open the file rename dialog, where you can change the name of the file.
- **Move / Copy File** will open a dialog, where you can choose where this file should be moved or copied
- **Delete** will open a dialog, where you can confirm the deletion the file

Folder context menu

The folder context menu can be accessed by right-clicking on a folder the perspective. It will offer some common folder management options.

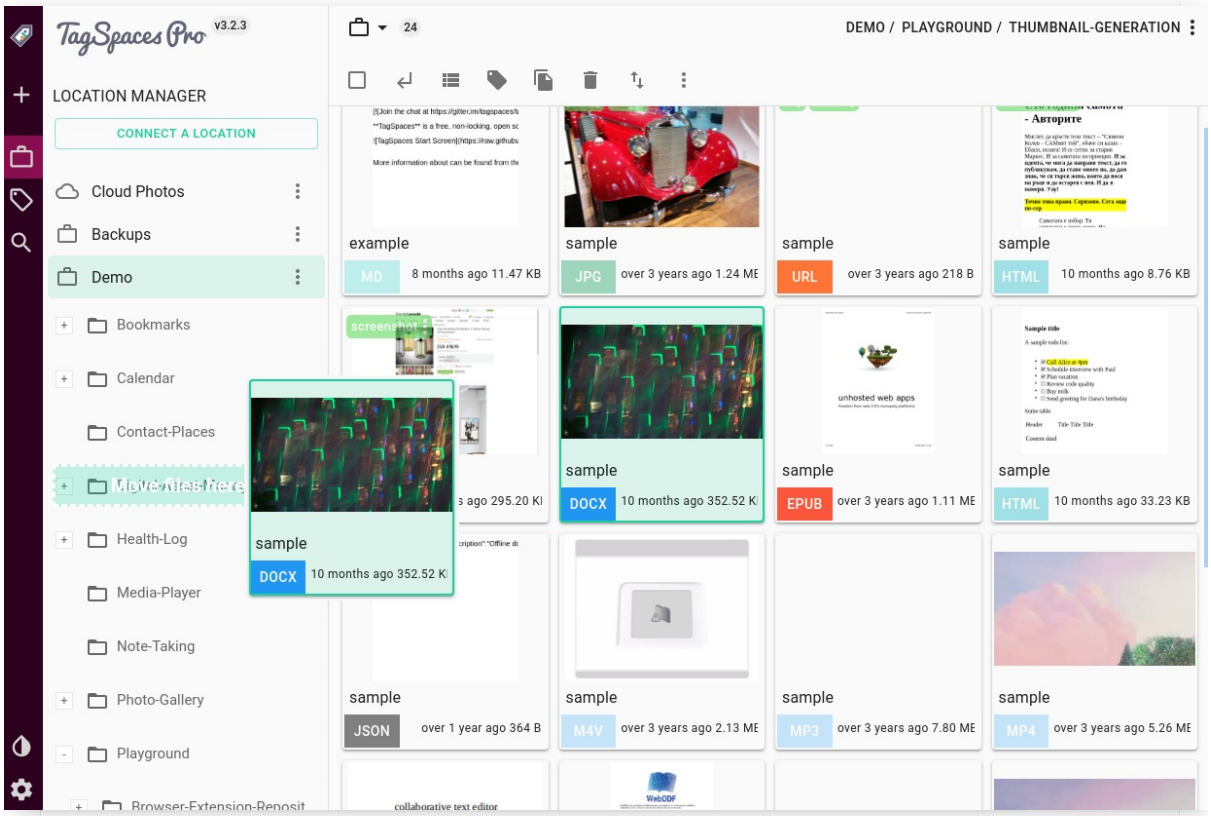


- **Open Directory** will navigate to the selected directory.
- **Rename directory** will open the directory rename dialog, where you can change the name of the folder.
- **Delete directory** will open a dialog, where you can confirm the deletion the directory
- **Show in File Manager** will open this directory in the default file manager of your operating system
- **Extract Content** will start the extraction of special data such as *geo-locations* from the files in this folder
- **Directory Properties** will open the properties of this folder in the preview area (most right panel) of the application

Drag to move within TagSpaces

An alternative way to move files into another folder is to drag it icon onto a folder on the **Folder navigation** area on the left panel. When the folder lights up with a greenish hue, just release the dragged item, and the file will be immediately moved into that folder.

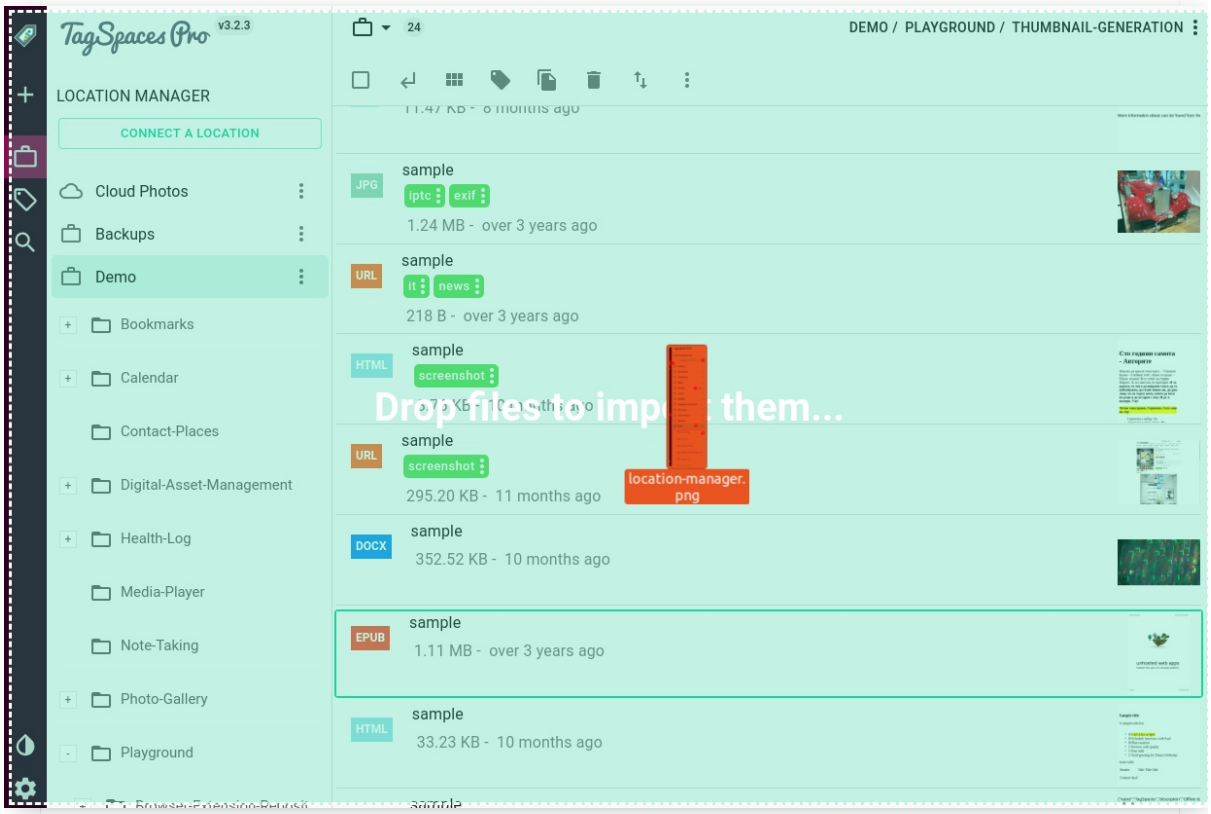
> **Hint:** You can access subfolders of any folder displayed in the hierarchy, by clicking the black folder icon next to its name.



Importing files with drag and drop

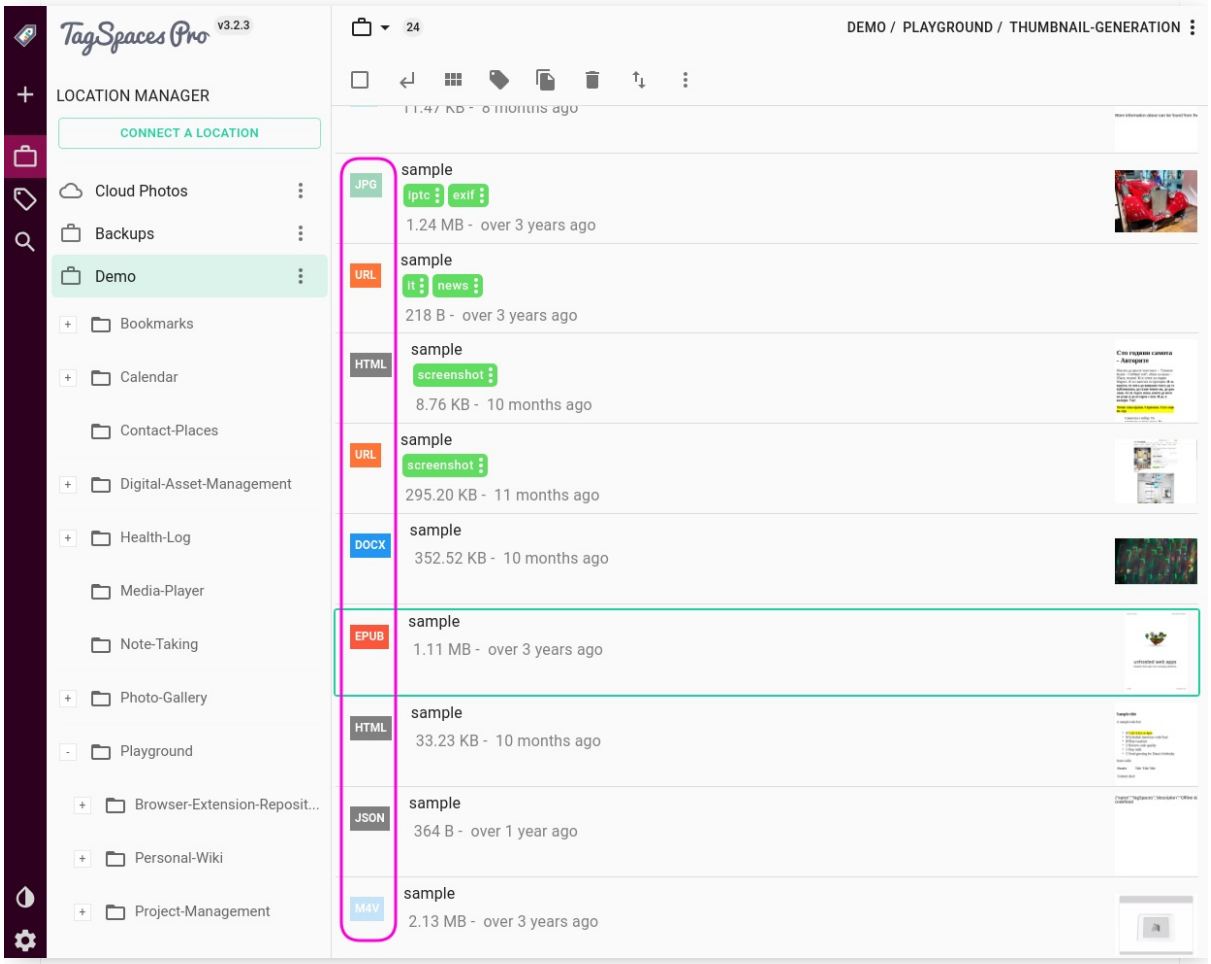
Besides dragging files from the **File Browser Area** to a sub folder, TagSpaces allows you to add files the currently opened folder by dragging and dropping it from the operating system, or its default file management application.

To do this, just grab a file icon with your mouse, and drag it to TagSpaces' application.



Color coded file extensions

In the grid and list perspectives, color coded file extensions are supported, allowing for a better visual recognition of the file type. For the most common file extensions, custom colors have been defined, to be displayed on the extension buttons found on the file rows or cards.



> **Hint:** The colors of the supported file types can configured in the **File Types** of application settings dialog.

Folder Visualization Perspective

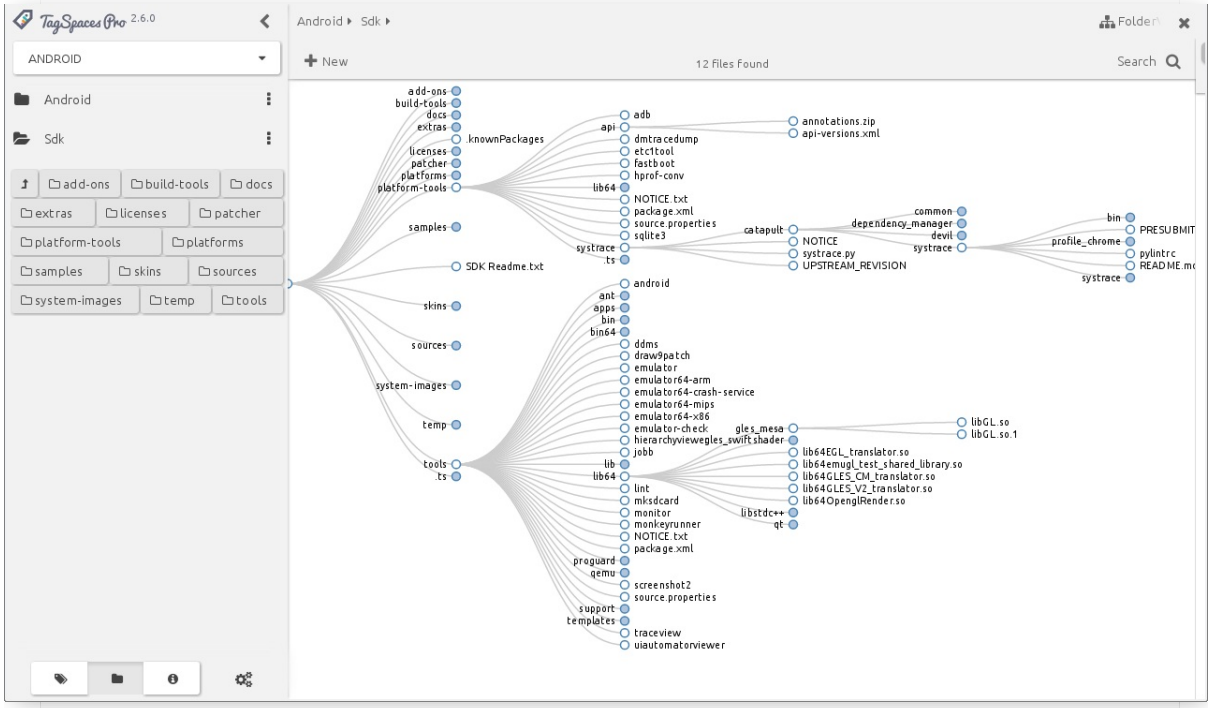
> **Note**: This perspective is not available in version 3 of TagSpaces.

The **FolderViz Perspective** is a collection of experimental views hat apply some information visualization concepts to representing your files and folders. While the primary aim of FolderViz perspective is to showcase TagSpaces' capabilities to developers, some of the options might offer some useful features to the end users.

FolderViz Perspective was offering five types of views, which were accessible from the FAB menu:

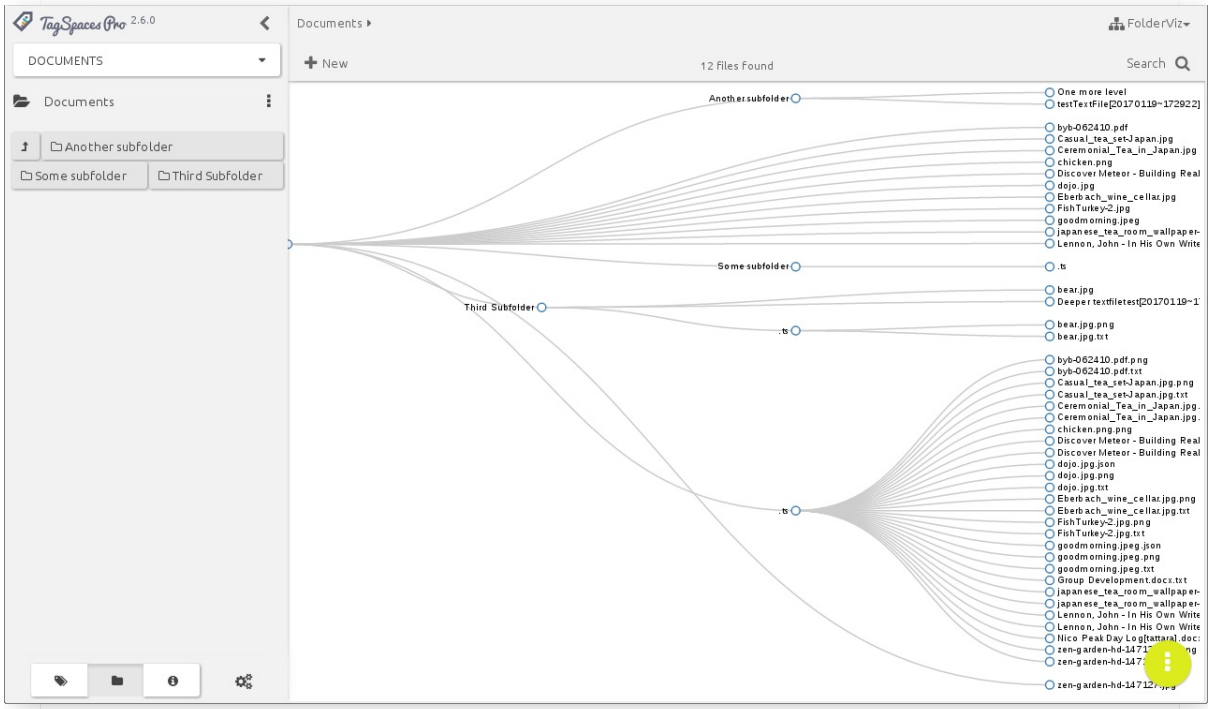
MindMap View (discontinued)

MindMap will display all folders and subfolders contained within the current directory, in an expandable tree-node format. There are plans to extend this view in the future with an inverted graph of tags so you can navigate your tagged files, with the help of the tag group three instead of the folder tree, and drag and drop files from one branch of tag tree to some other branch, which will automatically re-tag them.



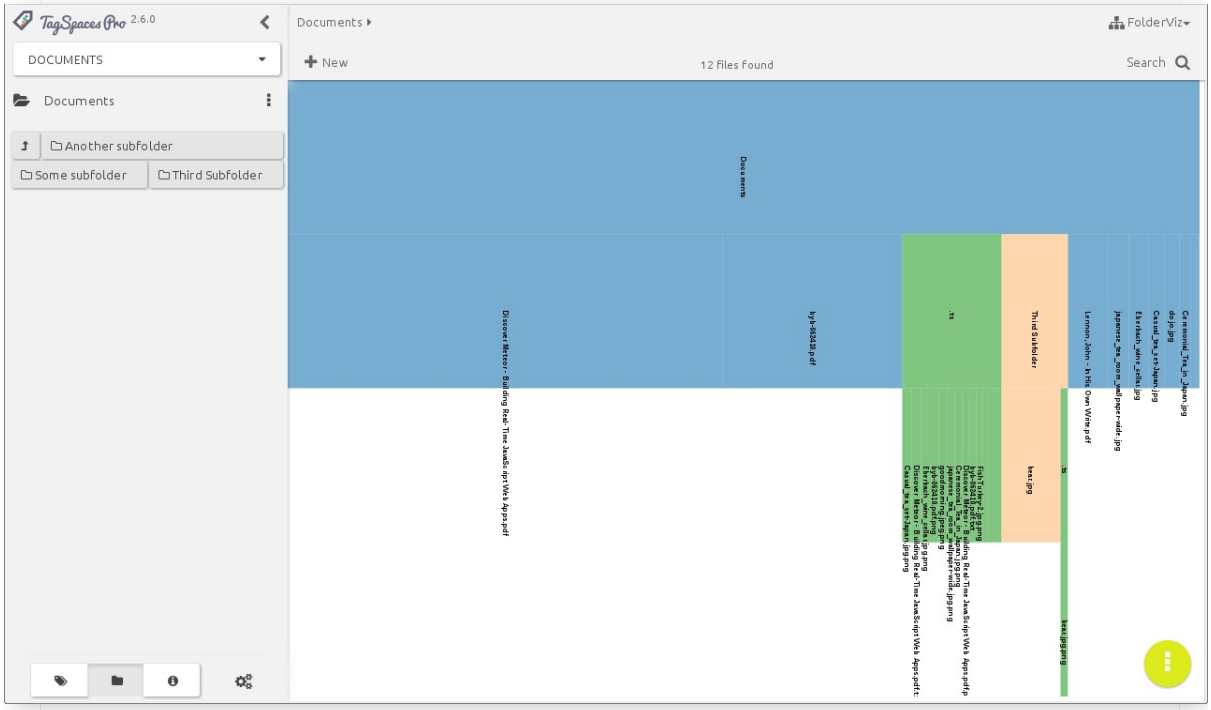
Tree View (discontinued)

Tree will display a fully expanded tree, similar to **MindMap**, only not interactive. It can be useful to visualize folder hierarchy, but this view can have performance issues with large directory structures.



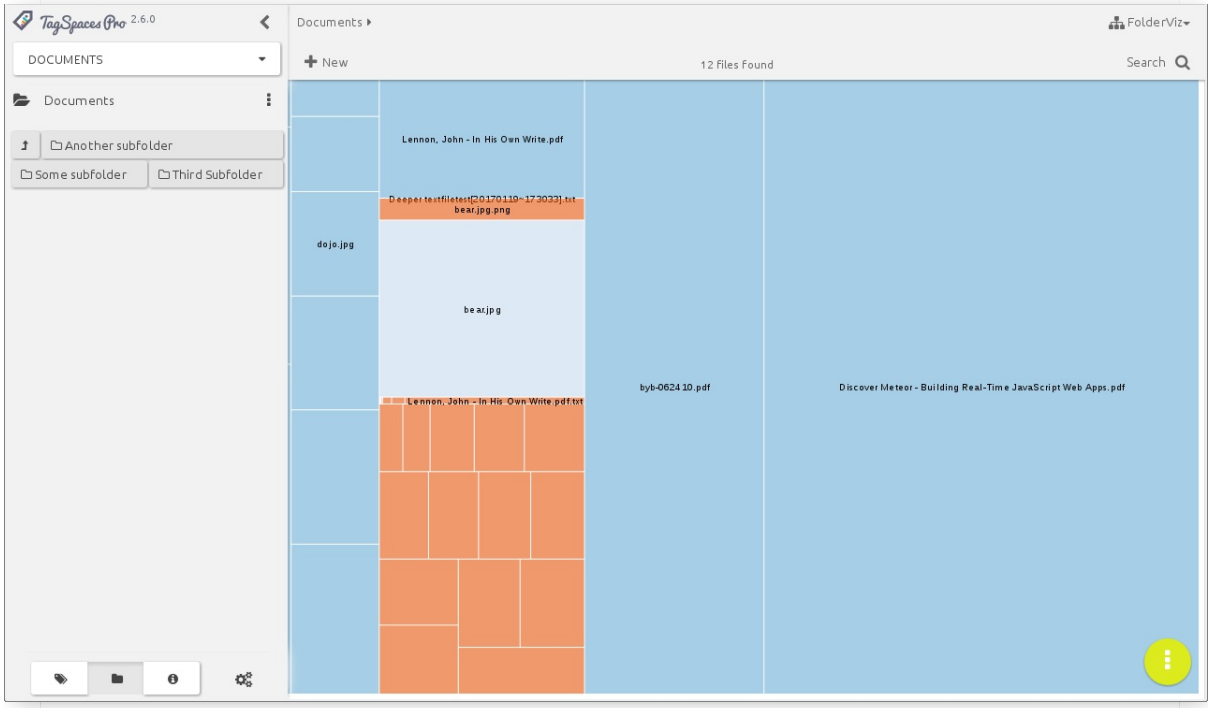
TreeMap View (discontinued)

TreeMap offers a representation of all files and folders, where the size of the squares correspond to file sizes, relative to the root and each other, while the structure of the squares represents folder hierarchy.



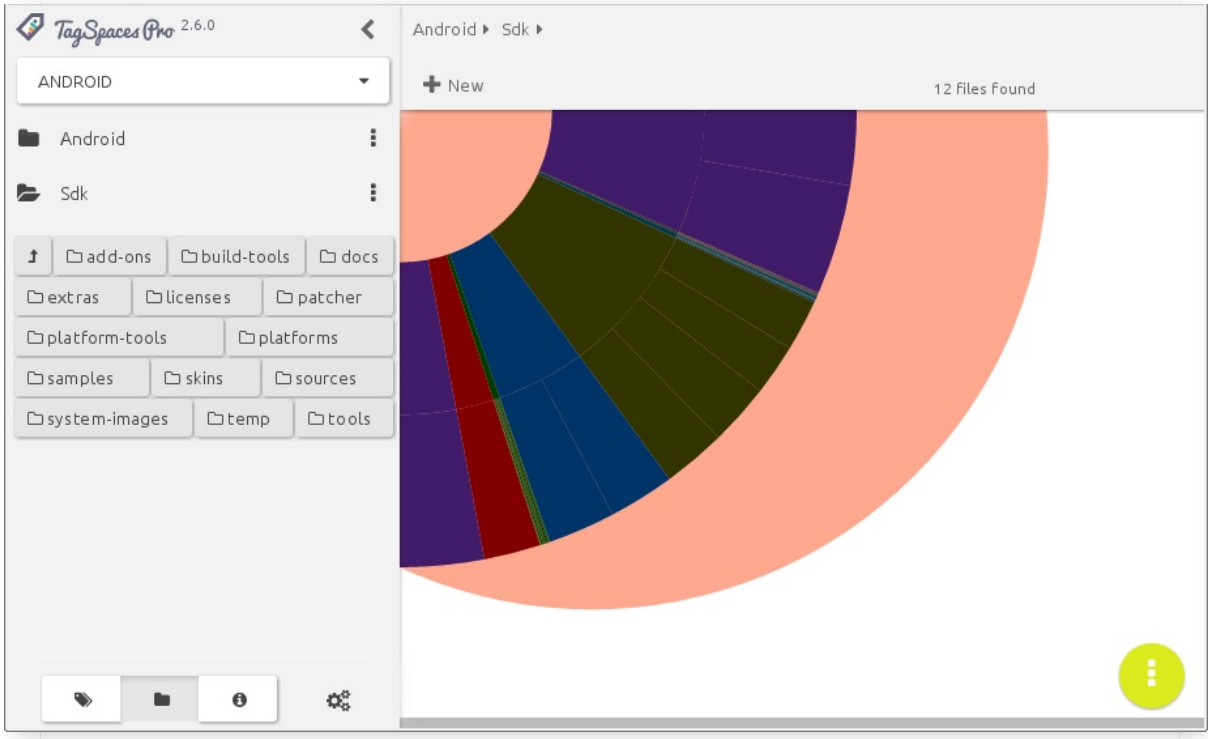
TreeMap-Navi View (discontinued)

TreeMap Navi is just like TreeMap, but without the hierarchy. Here the squares fully use up the available User Interface, allowing for a better visual representation of relative file sizes. This can be useful for finding large files or folders.



Bilevel Partition (discontinued)

Bilevel Partition is the most experimental and least functional of all views, basically a test to push the capabilities of folder and file visualization. While it might produce some interesting looking results, it will most certainly be removed from a future release of TagSpaces.



File preview and print

File preview

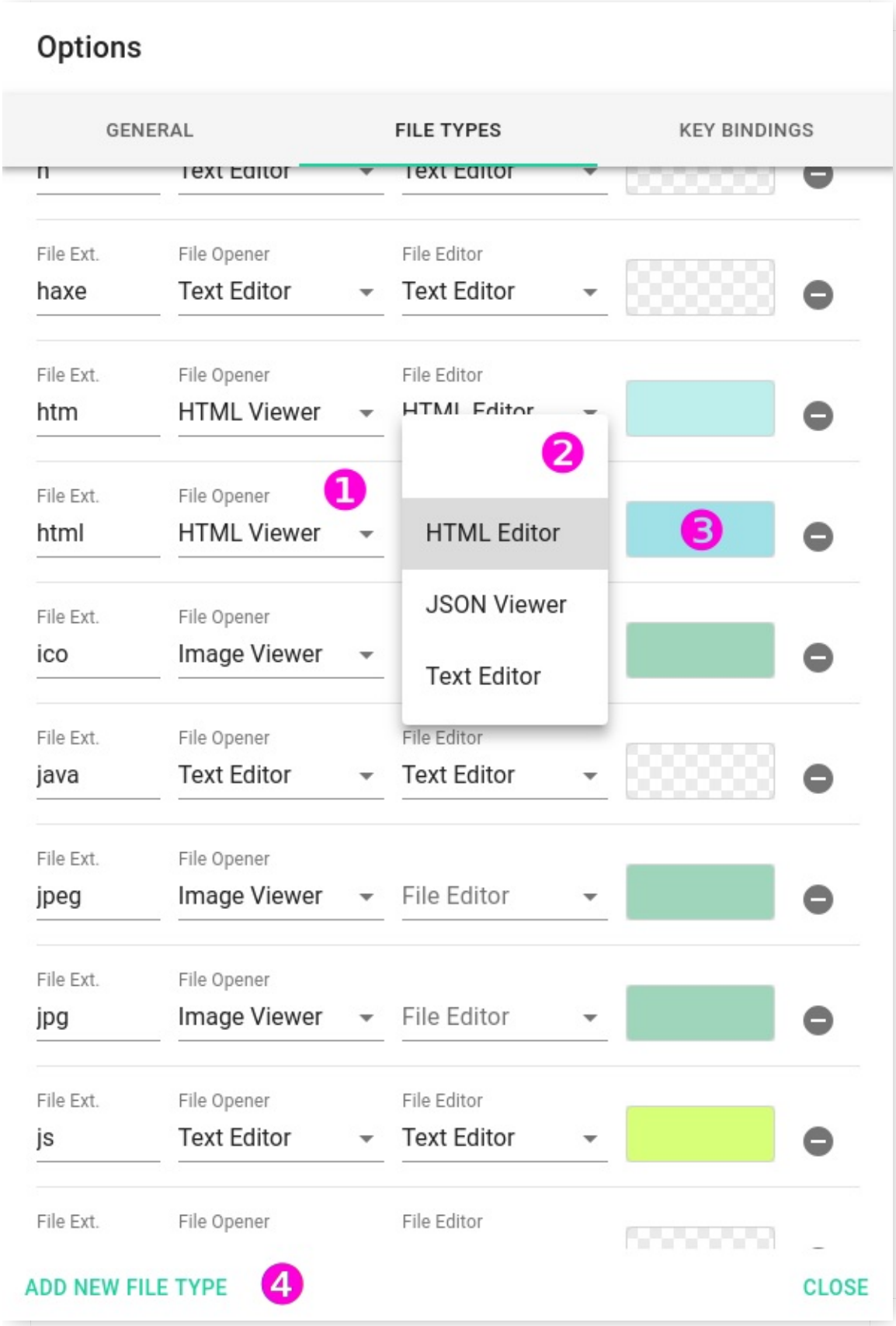
The application supports previewing of many file types without the need of external viewer. It comes with the following viewer extensions:

- Audio Video Player
- HTML Reader
- Image Viewer
- Link Opener
- Markdown Reader
- MHTML Reader
- PDF Viewer
- Text Reader
- Simple Viewer
- ZIP Opener

TagSpaces is designed with extensibility in mind so any other kind of file viewers can be easily developed and integrated.

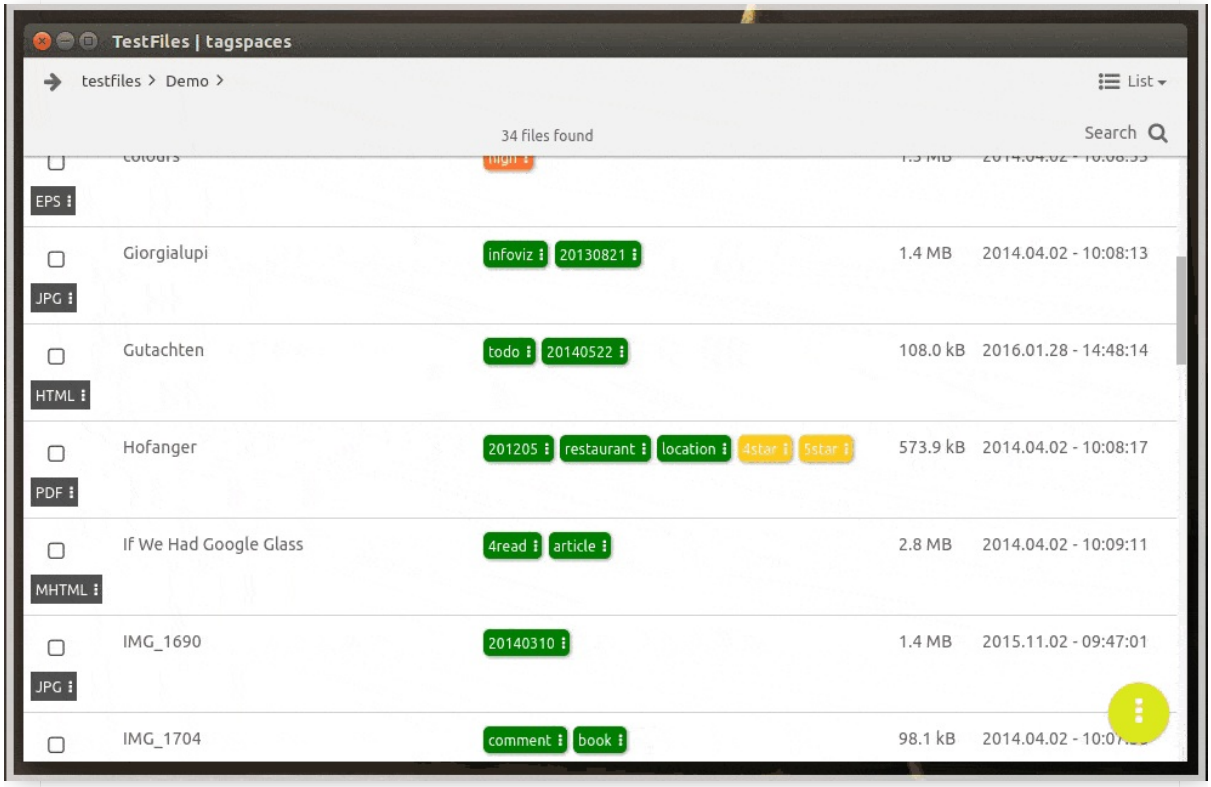
> **Source code browser and editor** The text editor supports source code highlighting for many common programming languages. This in combination with the build in JSON editor makes the application a good source code navigator with basic editing capabilities. The intention here is not to makes TagSpaces your next IDE, but rather to give your quick overview of source code repositories.

Assigning file viewer to file extensions



Printing files

The majority of the viewer extensions has tha ability to print the opened files, thanks to the build in print functionality. In the following short video, you can see how you can start the printing.

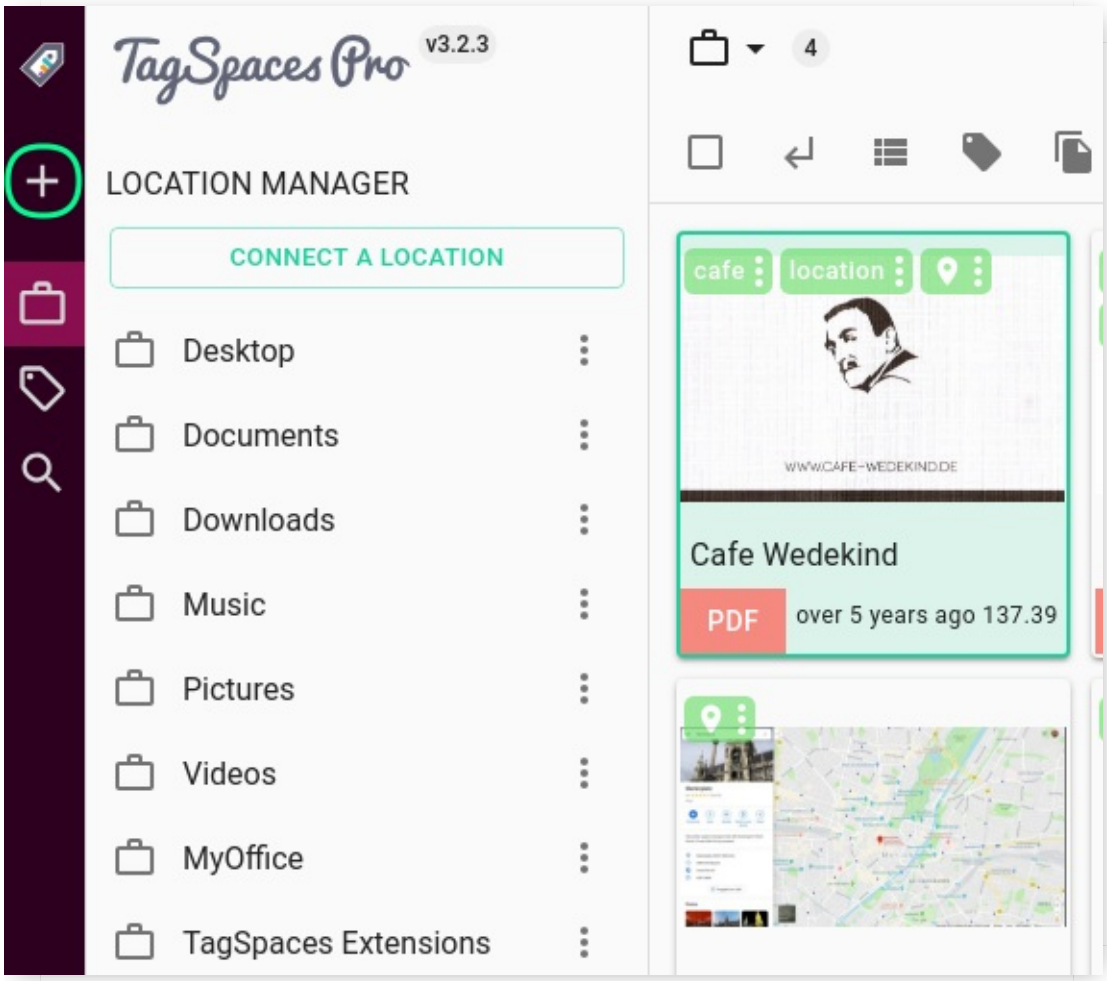


Crating and Editing files in TagSpaces

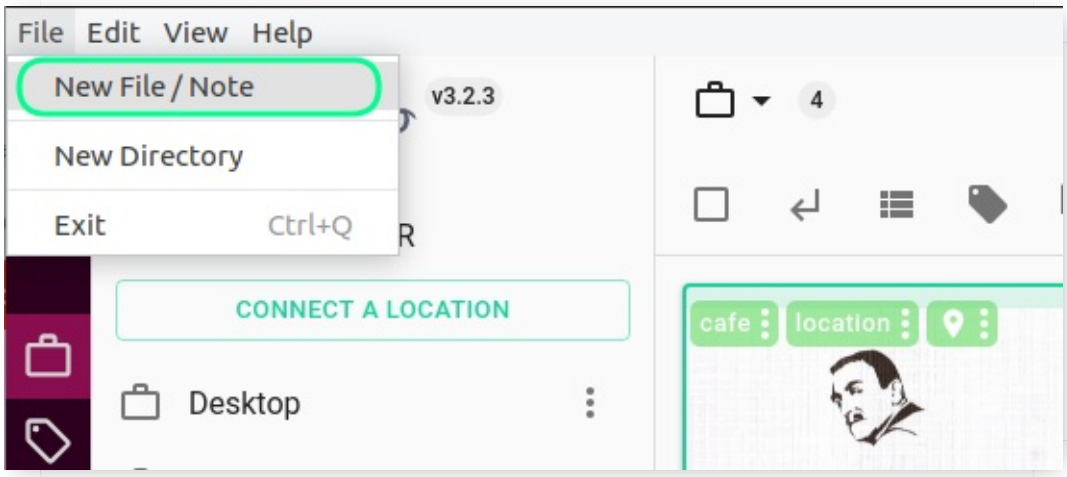
Creating files

Besides opening and previewing various file types, TagSpaces can also create new files of some text-based formats. File creation is a functionality that allows TagSpaces to become an effective note-taking application.

To create a new file in TagSpaces you have many options. The easiest one is from the button with plus icon, which almost always visible in the application



On the desktop version of the app you can also use the *File menu* from the *Main menu*. If the main menu is not visible you can open it by pressing the ALT keyboard key.

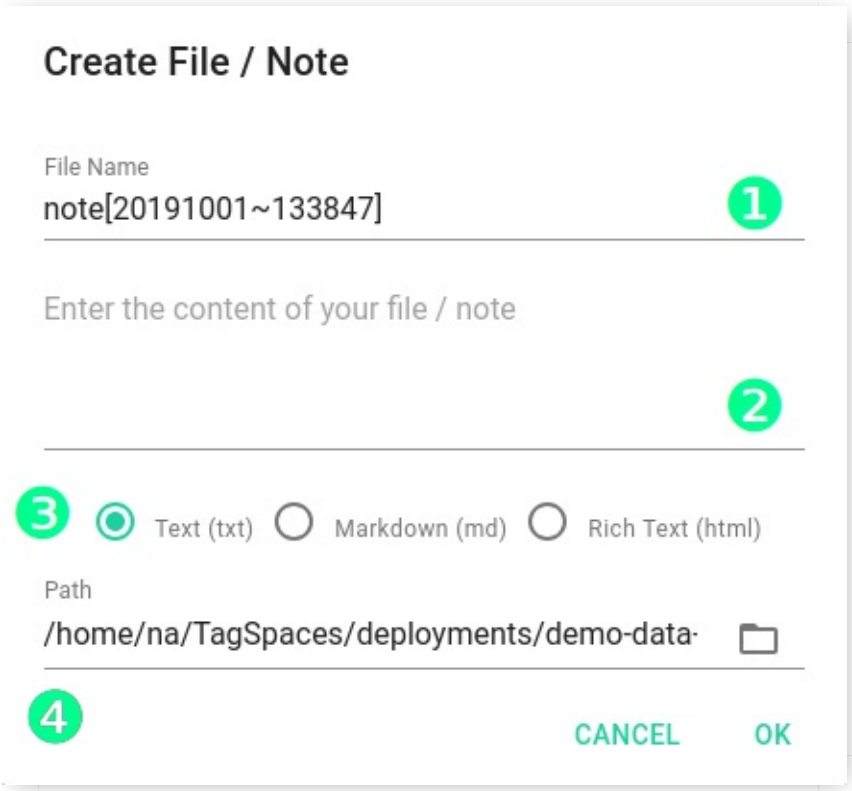


The triggering of the file creation action will open a *Create New Document* dialogue, where you can adjust the file name of the file, which will be create in area (1). In are (2) you can enter the text context of the file, you can copy and paste plain text, markdown or even HTML content here. This area could be left also empty allowing you enter content later. In a area (3) you should specify the type of the text content. Currently the following file formats are supported:

- **Plain text** - files created with this format will have the `.txt` file extension
- **MarkDown** - files created with this format will have the `.md` file extension. Learn more about the markdown

format here

- **Rich Text** - files created with this format will have the `.html` file extension.



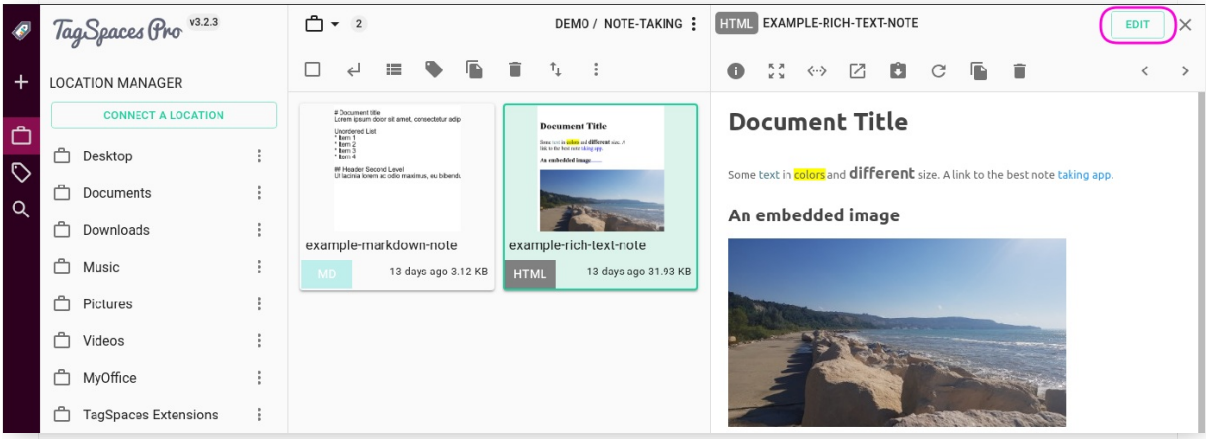
In the last section (4), you can specify where the file should be create. The initial path here is taken automatically from the currently opened folder.

Editing files

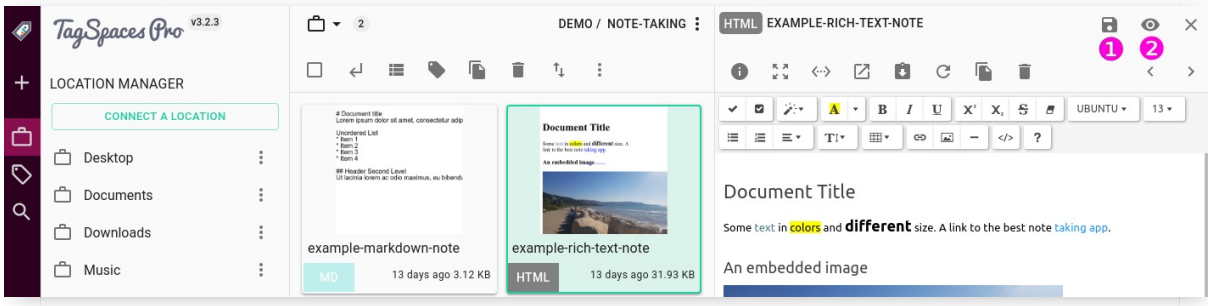
Besides opening and viewing files, TagSpaces can also edit specific file formats. Just like File Browser Perspectives, and File Preview extensions, the different types of **File Editors** are also **modular extensions**, making TagSpaces' file editing capabilities extendable. Currently three editor extensions ship bundled with the application:

- **HTML Editor** - `HTML` files serve a specific purpose in TagSpaces. They are treated as RichText documents, that can not only be previewed, but visually edited in a full-featured **WYSIWYG editor**.
- **Text Editor** - This editor opens all other text-based file formats, and unknown file types alike. For `.txt` and miscellaneous files, it behaves as a simple editor, with added syntax highlighting capabilities for common programming languages and a preview for **MarkDown**. (`.md`)
- **JSON Editor** - An interactive visual editor for editing and manipulating `JSON` files.

If the file format of the currently opened files is supported for editing the **EDIT button** will appear in the top right corner of the application. Clicking on this button will activate the edit mode for this file.



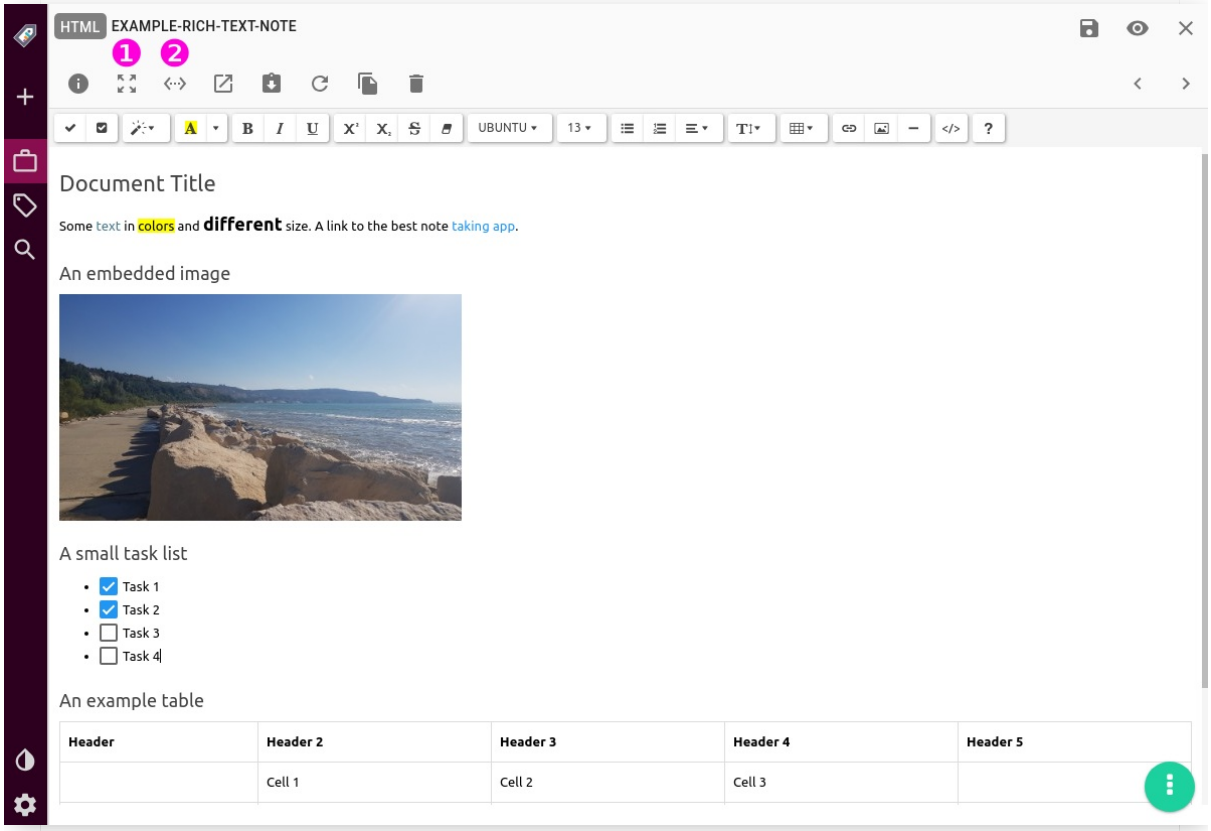
Once in the edit mode the **edit button** will disappear and on its place two new buttons will be visible. Pressing on button (1) will save the current changes. This can be done in the most cases also with the **CTRL+S** key combination. Pressing the button (2) will activate the preview mode by leaving the edit mode. If the changes were not saved, the app will ask you to save them.



Increasing your workspace

All of the above editors, will initially open on the right pane of the main area. You can expand your workspace to be able to see and edit more of the document, in two ways:

- **Expand** the editor to fill the whole width of the TagSpaces UI, by pressing the **Expand button** (2) as shown in the next screenshot.
- Pressing the **Fullscreen button** (1) will open only the area of the editor in full screen mode. To exit fullscreen mode, press the (X) button located at the top right or just press the **ESC** key on your keyboard.



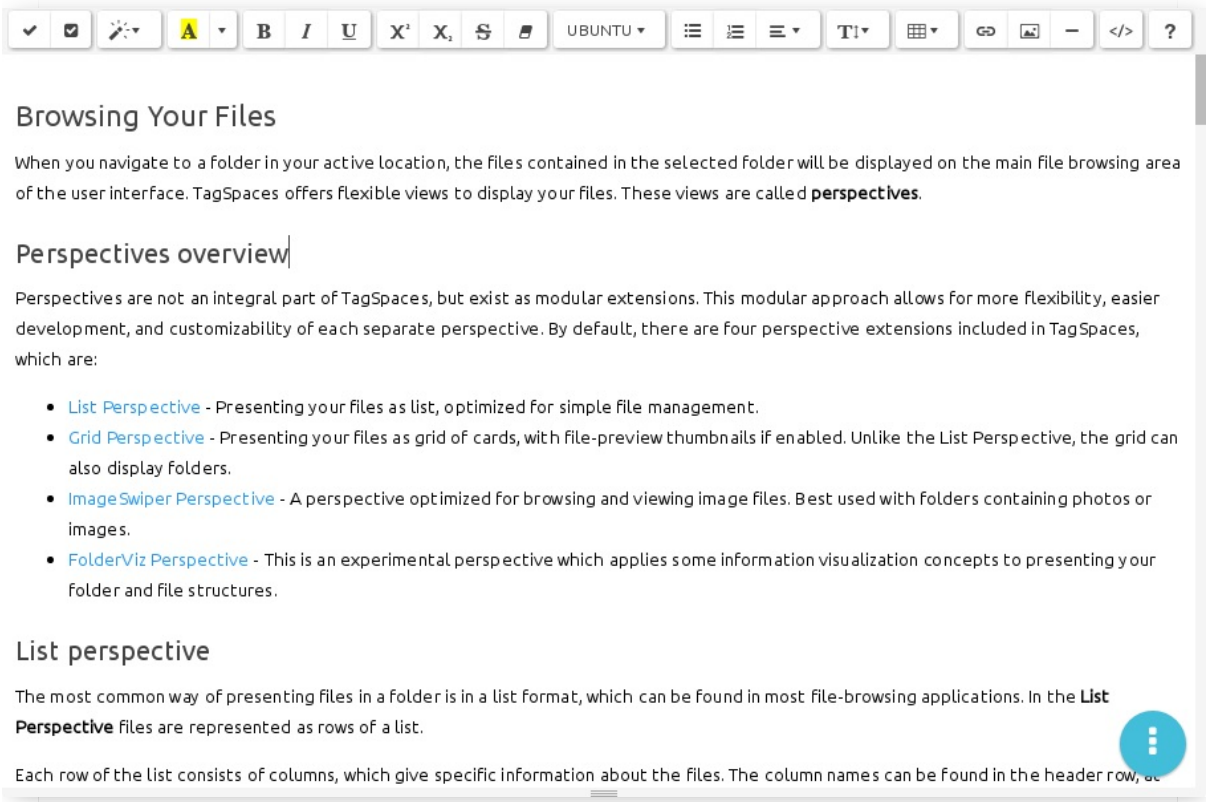
Distraction free editing

By using the fullscreen button you can edit documents in the so called **distraction free mode**. This will expand the editor area to fill the entire screen area, with no window decorations, or other UI elements, while any applicable **formatting toolbars** and the **Floating Action Button** will still be shown.

HTML Editor

HTML files are treated like RichText documents in TagSpaces. When you edit a `.html` document, the fully formatted preview of the file will be replaced by a **WYSIWYG** (**W**hat **Y**ou **S**ee **I**s **W**hat **Y**ou **G**et) HTML editor.

In editor mode the HTML document will keep its formatting, but you are now able to edit the text, making the HTML editor behave like a RichText editor. On the top of the view, you will find a formatting toolbar, with with you can fully control the appearance of the document



The formatting toolbar offers some common features, you can find in any RichText or WYSIWYG HTML editor, with some features specific to TagSpaces. The overflow menu options from the HTML viewer are not implemented in the HTML Editor.

Toolbar Buttons

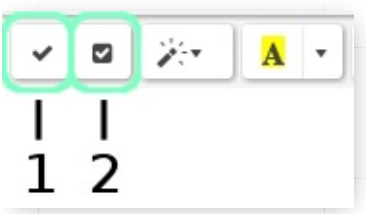
The toolbar buttons are grouped together based on similar behavior or functionality. The major groups are illustrated below

- **Manage Checkboxes** (**1**)
- **Paragraph Style** (**2**)
- **Color** (**3**)
- **Font Styles** (**4** and **5**)
- **Font Family** (**6**)
- **Lists, and Alignment** (**7**)
- **Line Height** (**8**)
- **Insert Table** (**9**)
- **Insert Link, Image and Horizontal Ruler** (**10**)
- **Code View** (**11**)
- **Help** (**12**)

Manage Checkboxes

This is a compound button, consisting of two parts:

- **Add checkbox** (**1**)
- **Toggle all checkboxes** (**2**)



Pressing **Add checkbox** will insert an interactive, clickable checkbox into your document, which can be used to create e.g. ToDo lists. To lean more about this feature, scroll down to the Creating ToDo Lists section.

Pressing the **Toggle all checkboxes** button will select or deselect all the checkboxes present in the HTML document, regardless of their location

> **Hint:** when some checkboxes are manually selectecd, while others are deselected, the *Toggle all checkboxes/ will check all the unchecked ones. Pressing the button again will only un-check the ones that have been checked via the button. the ones that were manually checked will remain unchanged.

Back to button group list

Paragraph Style

This button will open a drop down menu, from which you can select a paragraph style to apply. When selecting an option, the chosen style will be applied to the entire current paragraph (where the cursor is located), without having to make a selection first. If you do select some text first, the style will only be applied to the current selection.

Available styles are:

- **p** represents the `<p>` HTML tag (normal paragraph). It can also be used to remove other styles. The hotkey `Ctrl+0` is assigned to this action.
- **blockquote** will enclose the current paragraph into `<blockquote>` tags.
- **pre** will enclose the current paragraph into `<pre>` (preformatted text) tags.
- **H1** to **H6** mean different heading levels from **1** (largest) to **6** (smallest). Selection one of these options will enclose the entire paragraph into `<h1>` to `<h6>` tags. Hotkeys `Ctrl+1` to `Ctrl+6` can also be used to set headings, where the number reflects the level of the desired heading.

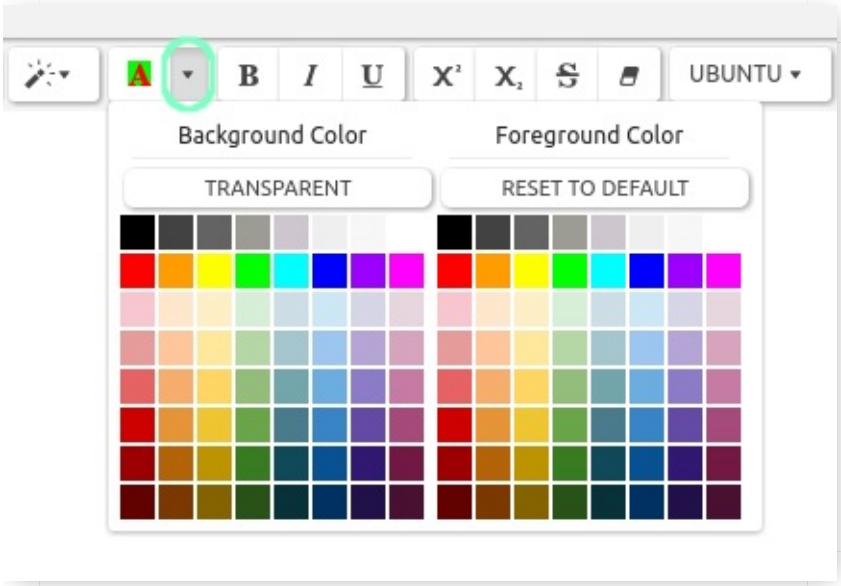
Back to button group list

Color

This is a compound button, which has two parts

The left part **Recent color** will apply the last used background and foreground color on the text. The applicable background and foreground colors are reflected on the button itself

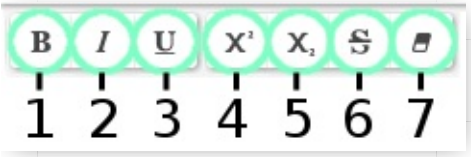
The right part **More Color** will open a dropdown menu, from which you can choose both background and foreground colors to apply, while the default color values can be reset with their respective buttons.



Back to button group list

Font Styles

The following groups represent font styles, that can be applied to either a selection, or as a toggle, to mark any text to be written with the given style, until the toggle is switched off.



Bold (**1**), **Italic** (**2**), and **Underline** (**3**) will mark either the selected text, or the text that follows as **bold**, *italic* or underlined, replectively. These three buttons also have hotkeys assigned. `Ctrl+b` will toggle **bold**, `Ctrl+i` toggles *italic*, while `Ctrl+u` toggles underline.

Superscript (**4**), and **Subscript** (**5**) will mark either the selection or the text that follows to be superscript or subscript respectively. **Strikethrough** (**6**) will create ~~strikethrough~~ text, and **Remove Font Style** (**7**) will remove all formatting. (This last option is only applicable to selections.)

From this group, only two options have hotkeys assigned. **Strikethrough** can be toggled with `Ctrl+Shift+s`, while the **Remove Font Style** hotkey is `Ctrl+\`

Back to button group list

Font Family

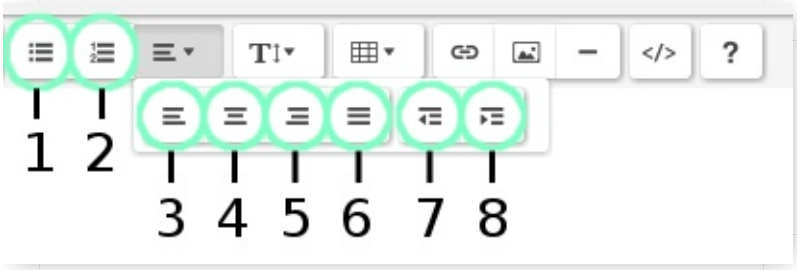
This is the last of the font style buttons. It can set the selection or the text that follows to a specific font family.

> **Hint:** The options presented here might depend on your operating system and the fonts installed

Back to button group list

Lists, and alignment

This is a compound group, which offers different options, that all work on the currently active paragraph, as a toggle, without the need to make a selection first.



The first two buttons offer toggles for **Unordered List** (**1**), and **Ordered List** (**2**). The last button in the group will open a dropdown menu which offers four standard alignment options **Left** (**3**), **Center** (**4**), **Right** (**5**), and **Full Justify** (**6**); and also the option to **Decrease Indent** (**7**), or **Increase Indent** (**8**).

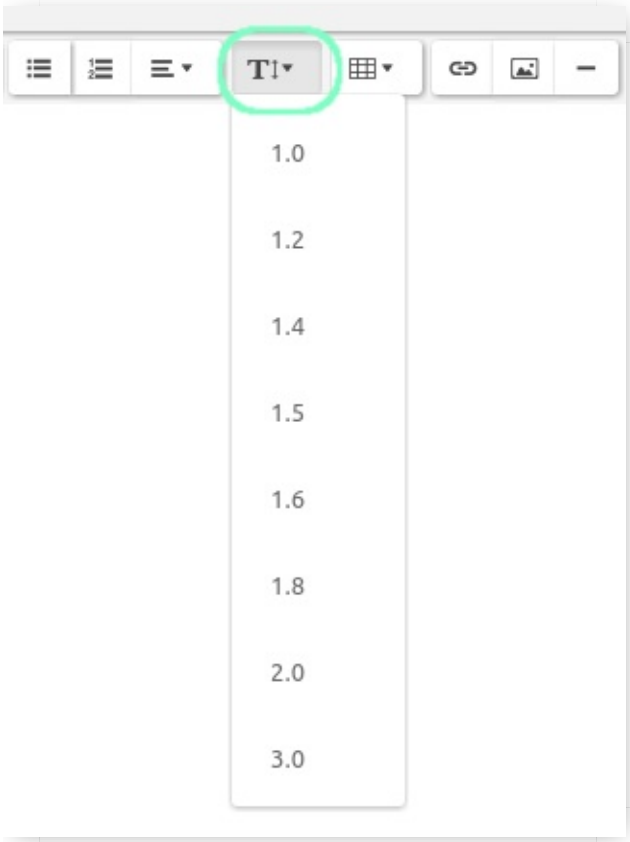
The hotkeys for these operations are:

- **Unordered List** - `Ctrl+Shift+7`
- **Ordered List** - `Ctrl+Shift+8`
- **Left align** - `Ctrl+Shift+L`
- **Center align** - `Ctrl+Shift+E`
- **Right align** - `Ctrl+Shift+R`
- **Justify full** - `Ctrl+Shift+J`
- **Decrease indent** - `Ctrl+Shift+Tab`
- **Increase indent** - `Ctrl+Tab`

Back to button group list

Line Height

Offers a dropdown menu that allows you to set the line-height of either the active paragraph or the currently selected paragraphs between **1.0** and **3.0** by increments of **1.0**, **1.2**, **1.4**, **1.5**, **1.8**, **2.0**, and **3.0**

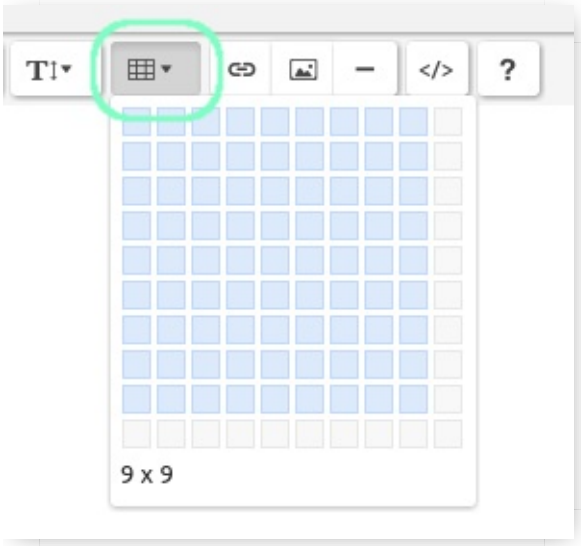


> **Hint:** When you make a selection, you do not need to select the entire paragraph. The selection's edges will mark active paragraphs. The line height adjustment will be performed on all full paragraphs, starting with the one inside which the selection starts, and ending with the one, inside which the selection ends.

Back to button group list

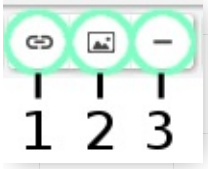
Insert Table

This button will open a drop down graphical menu for drawing a table grid, with a maximum size of **10x10**, that can be easily inserted into the document



Back to button group list

Insert Link, Image, and Horizontal Ruler

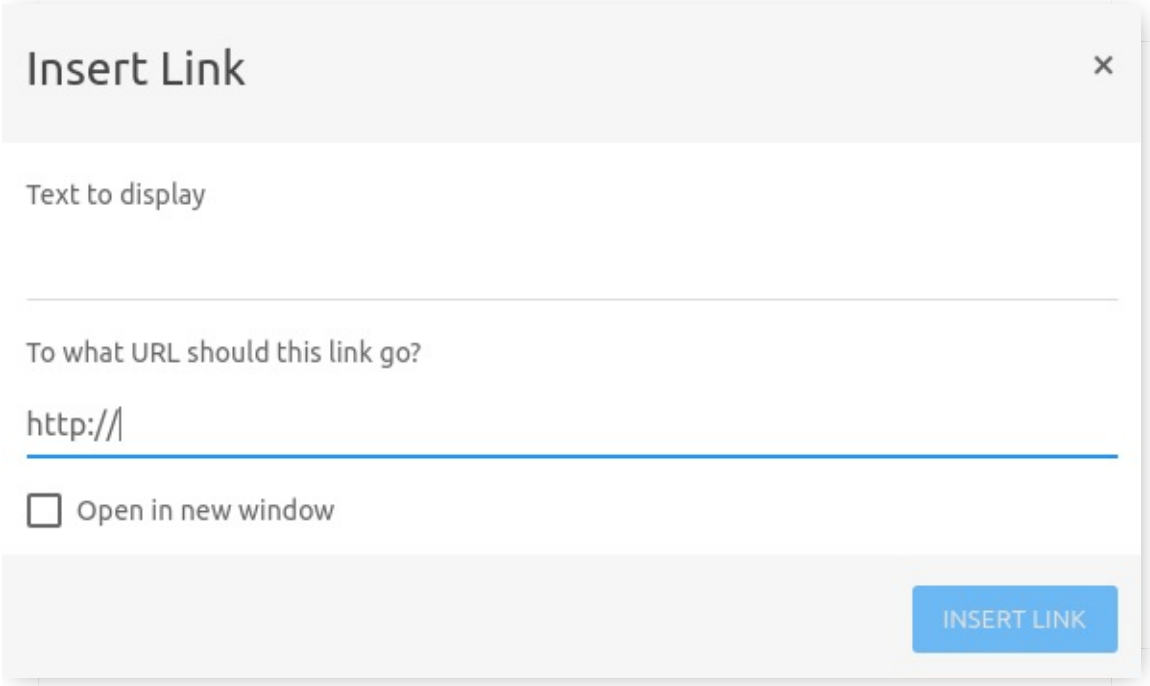


This group has the following three buttons

- **Insert Link** (**1**)
- **Insert Image** (**2**)
- **Insert Horizontal Ruler** (**3**)

Insert Link

The first button will open a dialog, which allows you to add a hyperlink to the text. The dialog allows for specifying the text to display, and the link to follow, when clicked, with a checkbox allowing to set the link to be opened in a new tab or window. The dialog can also be invoked by pressing `Ctrl+K`



If you select some text to apply the link to, before pressing the button, the selection will automatically populate the *Text to display* field. If you made no selection, you will need to specify a text to be displayed. If you leave the field empty and start typing a URL, it will automatically populate the text to display field, which you can later modify.

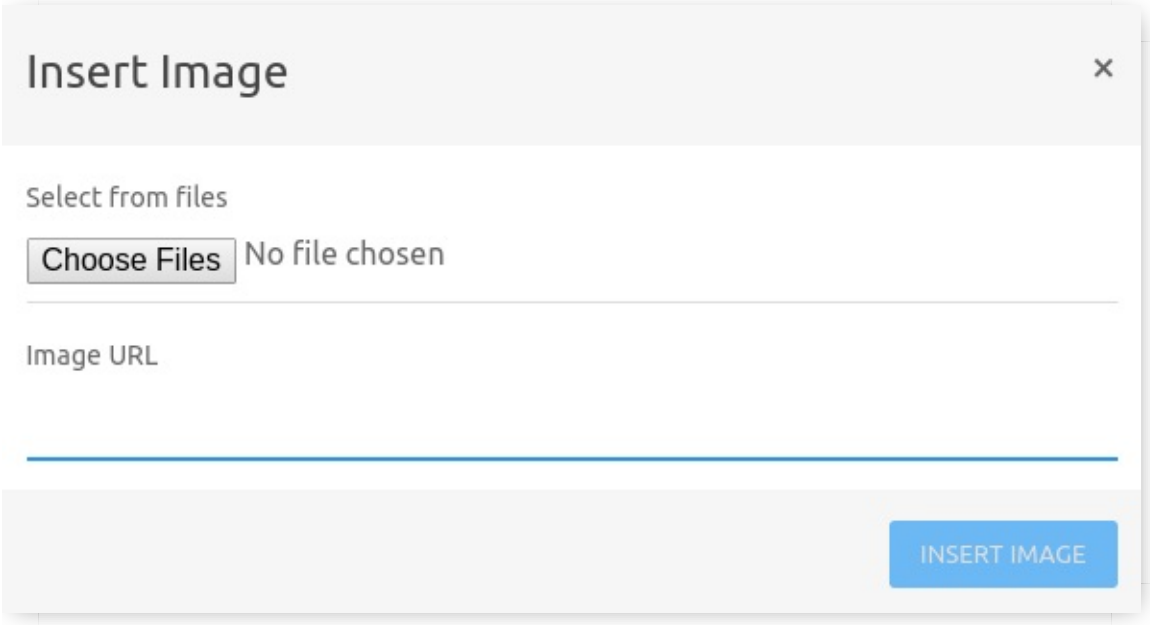
To edit or remove links, you can just click inside the link, and use the buttons that appear on the popup.

The first button (**1**) will allow you to edit the link, while second (**2**) will remove it.

> **Hint:** Links can also be added to the documents by typing the URL, and pressing enter. A well formed URL will automatically turn into a link.

Insert Image

The second button in the group will also open a dialog, which lets you insert a picture from either your computer, or from a URL.



Insert Horizontal Ruler

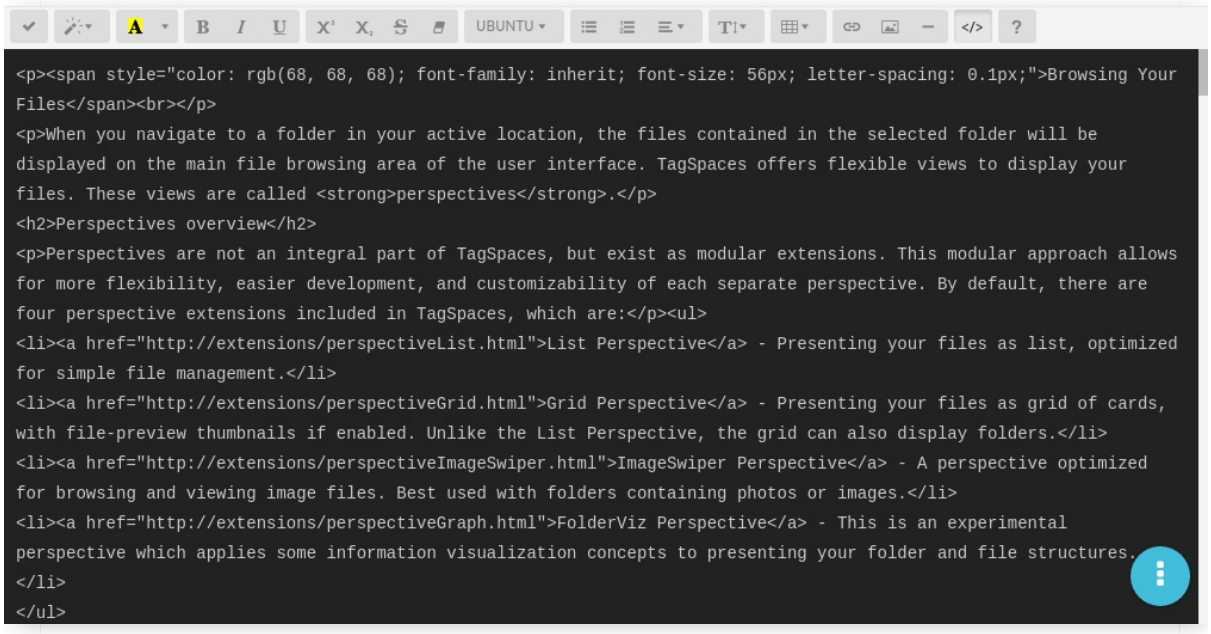
The last button will insert a simple horizontal ruler, (a HTML `<hr/>` tag) at the point where the cursor is currently located. The hotkey to quickly insert a horizontal ruler is `Ctrl+Enter`.

> **Hint** This button does not respect paragraph endings. If you press this button at the middle of a paragraph, it will break the paragraph at that point.

Back to button group list

Code view

This is a toggle to show plain HTML code on a dark background, where the formatting buttons are disabled, and you can edit the HTML code directly.



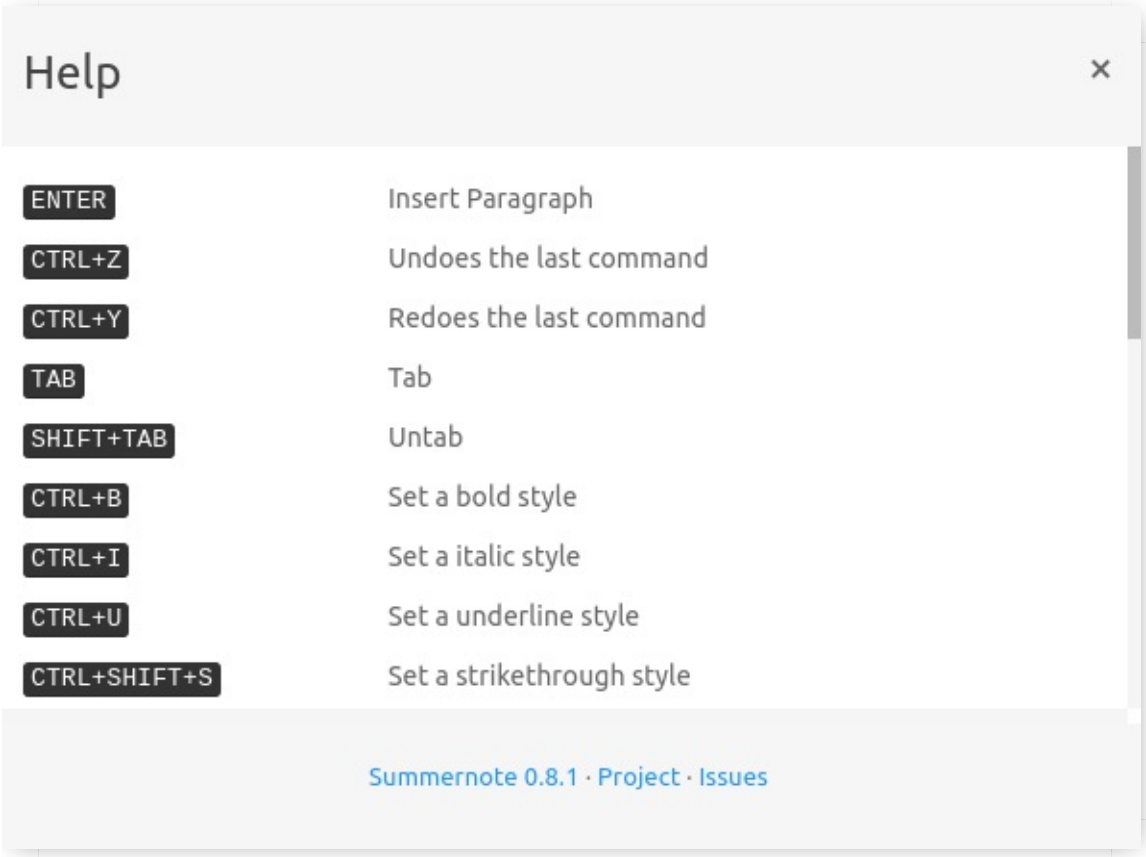
Pressing the button again, will return to the WYSIVYG mode.

> **Hint:** Code highlighting for the HTML view is not currently implemented.

Back to button group list

Help

The last button will display a summary of all the keybindings. If you prefer to use hotkeys, you will find a quick reference here.

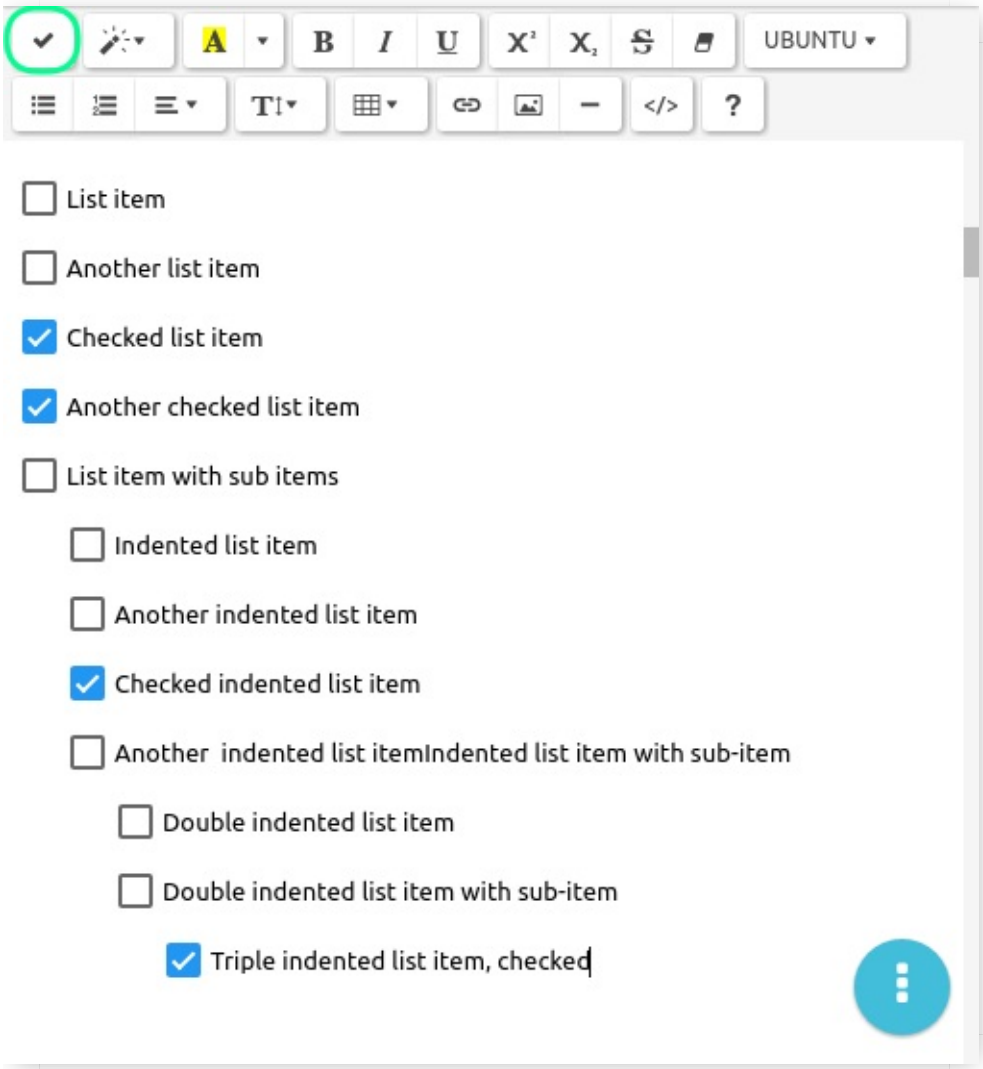


Back to button group list

Creating ToDo Lists

The recently added feature of interactive checkboxes makes TagSpaces ideal for creating simple, yet flexible **ToDo lists**. Pressing the **Add Checkbox Button** button will insert a checkbox anywhere in the text.

For **ToDo Lists**, you would ideally want the checkbox to be the very first character of a list, although you are not limited by placement. To make your list multi-level (i.e. sub-items under list items), you can use the **Increase Indent** feature (see Toolbar Buttons -> Lists and Alignment) on the desired line, by either pressing its button or using its hotkey (`Ctrl+Tab` to increase indent, `Ctrl+Shift+Tab` to decrease intent).



> **Hint** The interactive checkboxes only work in editor mode. When in HTML preview mode, you can see the current state of the box, but cannot change it. To make a list interactive, switch to editor mode, by pressing the little pencil icon at the top of the preview.

Key bindings

Below you can find a summary of all the hotkeys you can use in the HTML Editor

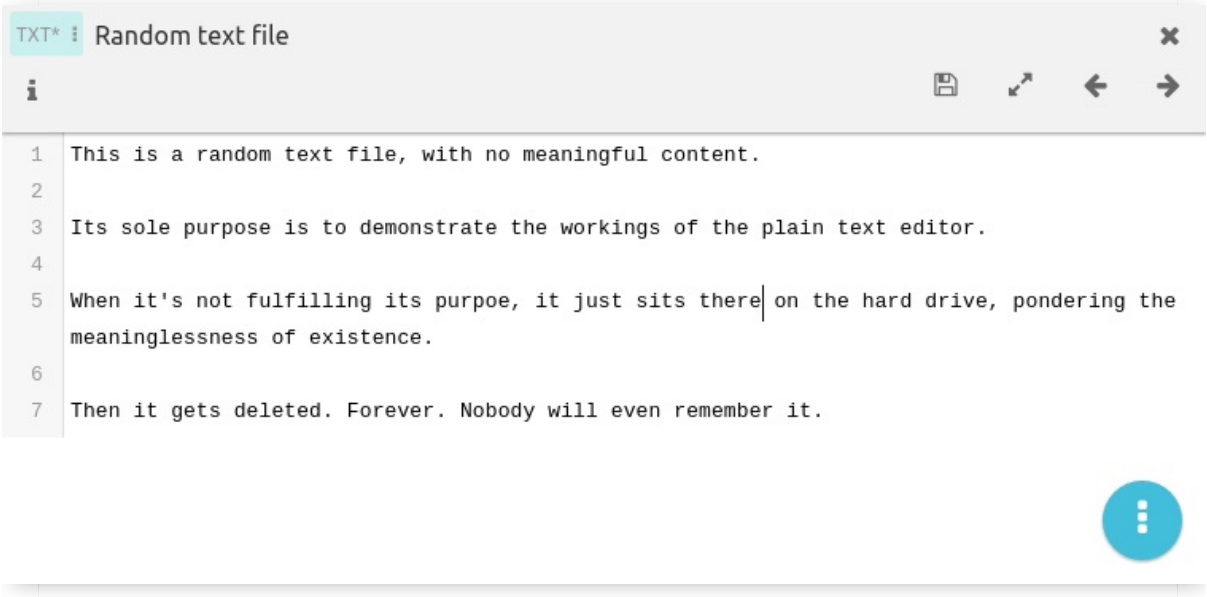
> **Hint:** Mac users should use the `Cmd` key, where `Ctrl` is mentioned

- `Ctrl+Z` - Undo the last command
- `Ctrl+Y` - Redo the last command
- `Tab` - Insert Tab
- `Shift+Tab` - Remove tab
- `Ctrl+B` - Set a bold style
- `Ctrl+I` - Set an italic style
- `Ctrl+U` - Set an underline style
- `Ctrl+Shift+S` - Set a strikethrough style
- `Ctrl+\` - Clear all styles
- `Ctrl+Shift+L` - Set left align
- `Ctrl+Shift+E` - Set center align

- `Ctrl+Shift+R` - Set right align
- `Ctrl+Shift+J` - Set full align
- `Ctrl+Shift+7` - Toggle unordered list
- `Ctrl+Shift+8` - Toggle ordered list
- `Ctrl+Tab` - Indent on current paragraph
- `Ctrl+Shift+Tab` - Outdent on current paragraph
- `Ctrl+0` - Change current paragraph's style to plain paragraph (`<p>` tag)
- `Ctrl+1` - Change current paragraph's style to H1
- `Ctrl+2` - Change current paragraph's style to H2
- `Ctrl+3` - Change current paragraph's style to H3
- `Ctrl+4` - Change current paragraph's style to H4
- `Ctrl+5` - Change current paragraph's style to H5
- `Ctrl+6` - Change current paragraph's style to H6
- `Ctrl+Enter` - Insert horizontal rule
- `Ctrl+K` - Show Link Dialog

Text Editor

When you edit a plain text file, or any unsupported file format that displays the edit button, the simple text editor will open.

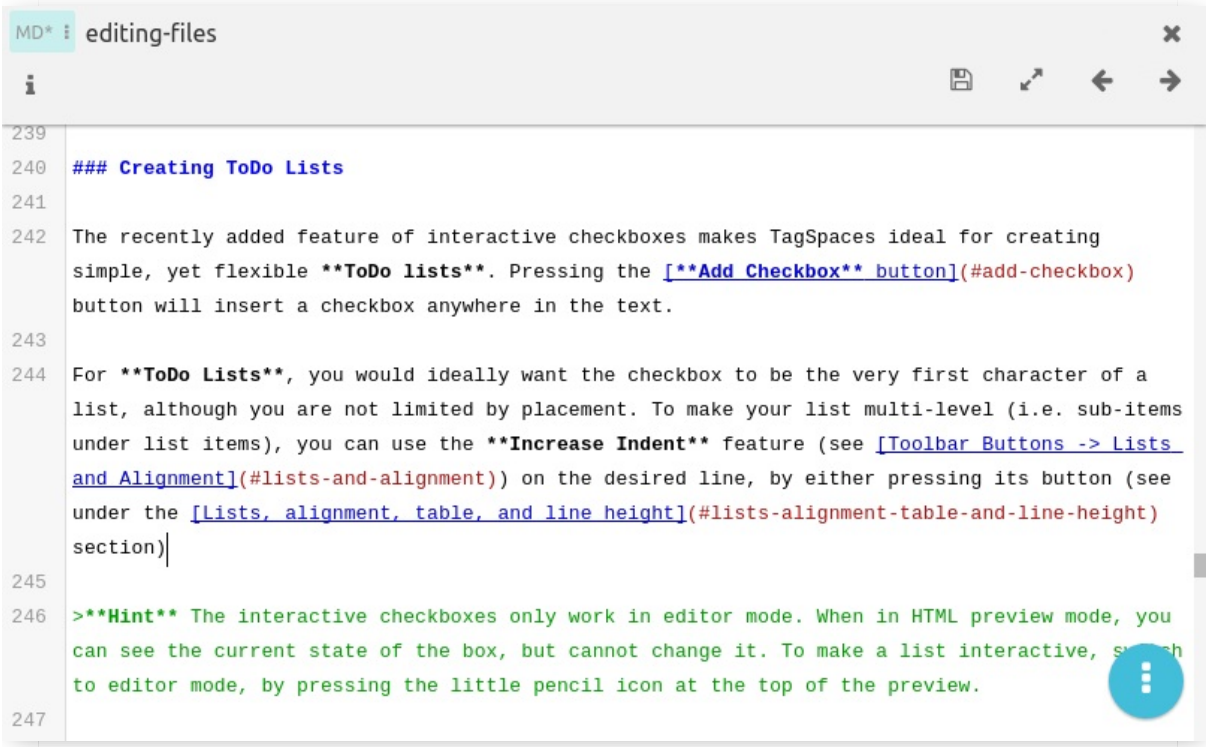


The only extra the text editor offers is basic code highlighting for some common programming languages, and numbered lines.

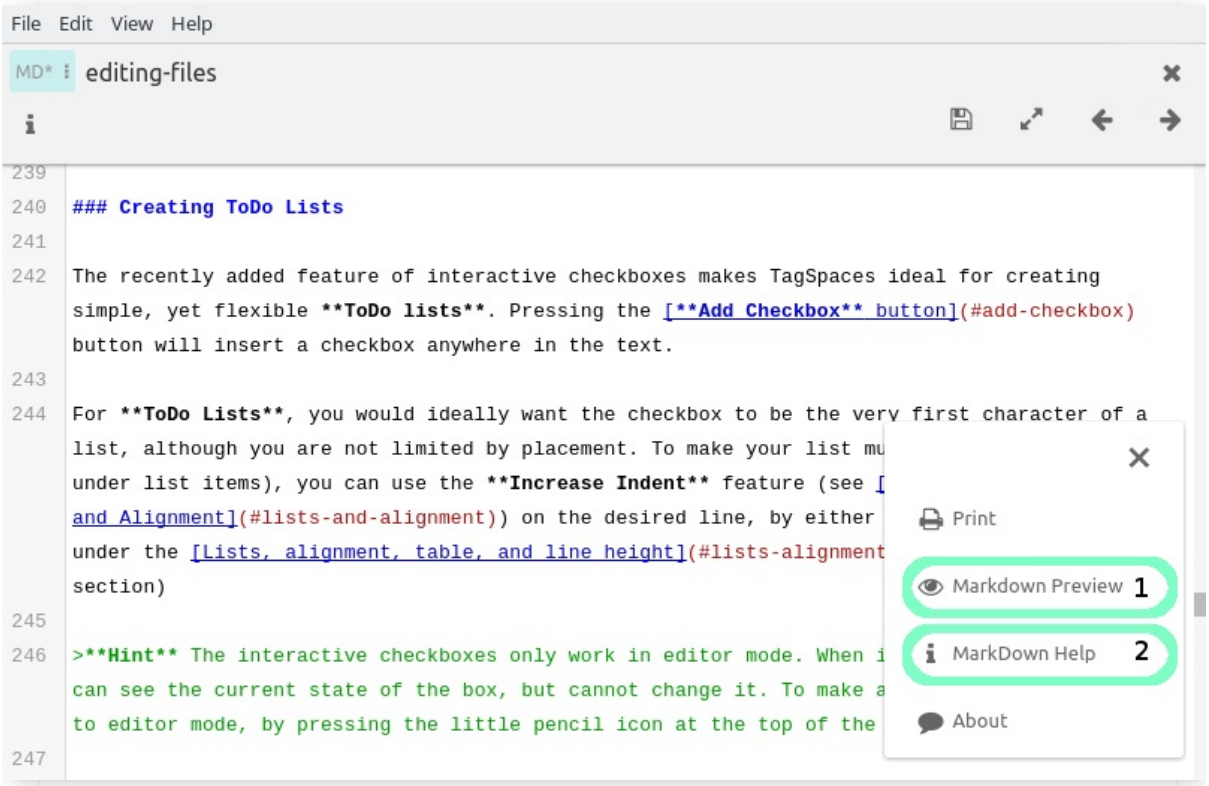
The text editor will keep the option to print from the plain text viewer.

MarkDown Editor

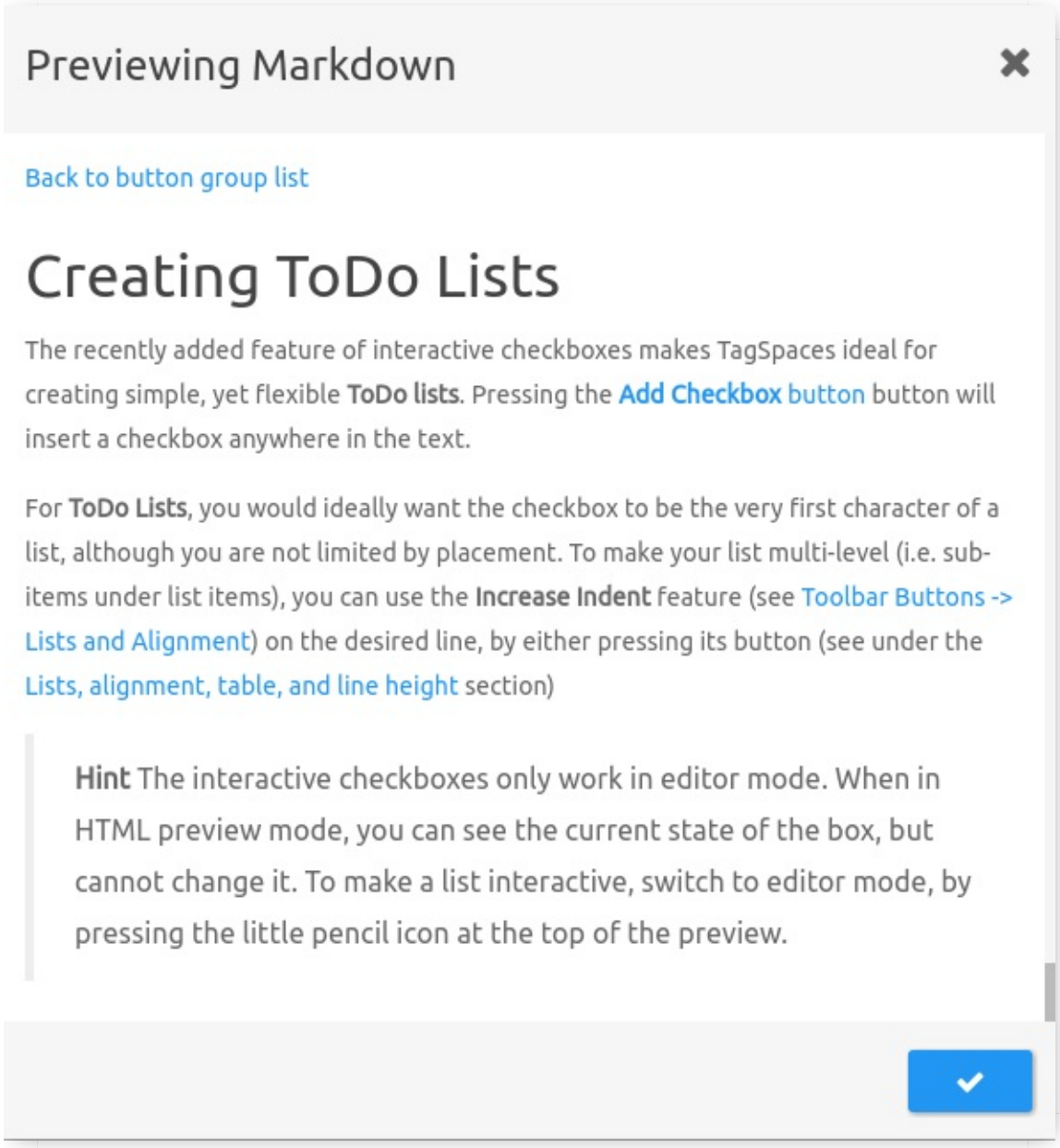
When you open a MarkDown file for editing from the MarkDown Viewer, it will be opened wit the same **Text Editor**, any plain text file would, offering inline highlighting of Markdown syntax.



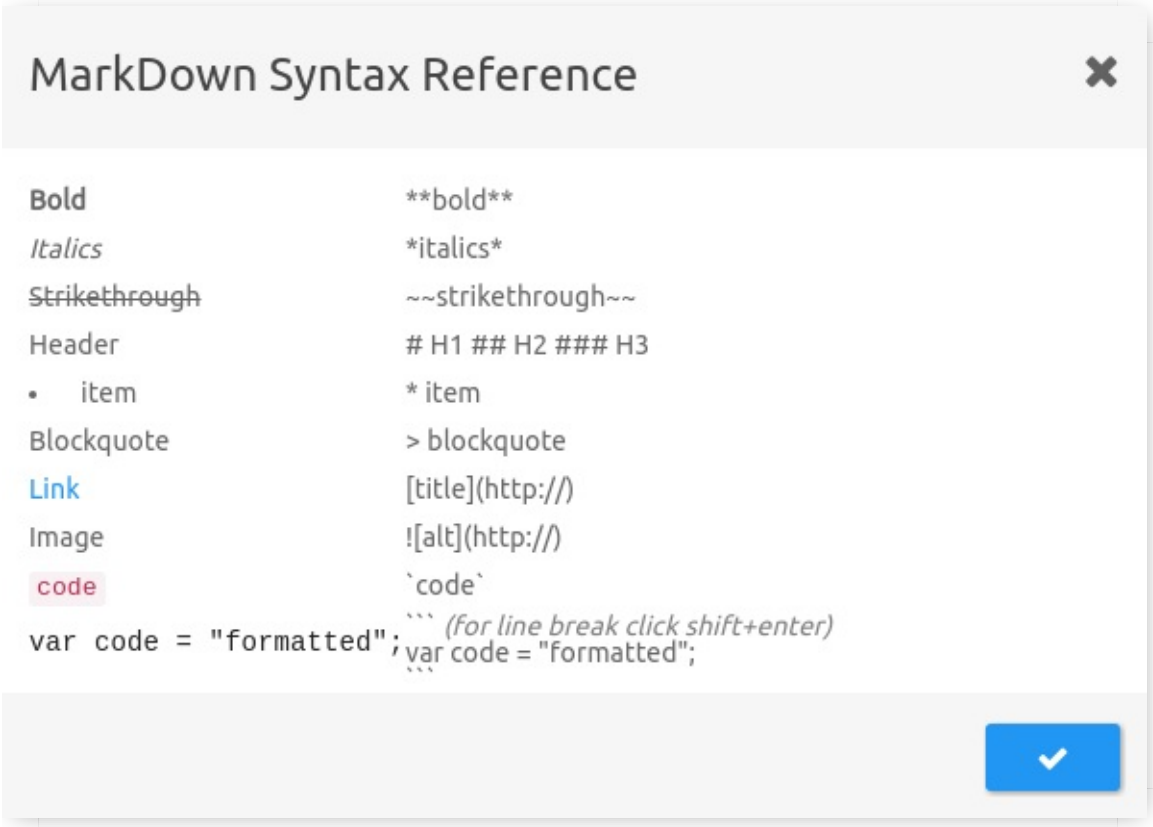
The difference between MarkDown and plain text editors is the two additional functions in its **FAB Oveflow Menu**: *MarkDown Preview* (**1**), or *MarkDown Help* (**2**).



The first option will open a popup window, presenting a formatted preview of the MarkDown file, similar to what you would see in the MarkDown viewer



The second option will offer some basic help about MarkDown syntax and formatting



Linking local files and images

Markdown editor will also allow you to **link local files and images**, from within your connected location, using a path, relative to your currently active folder. **Linked local images** will be shown embedded in the current markdown preview, while **linked files** will open in the default external application, defined by your operating system.

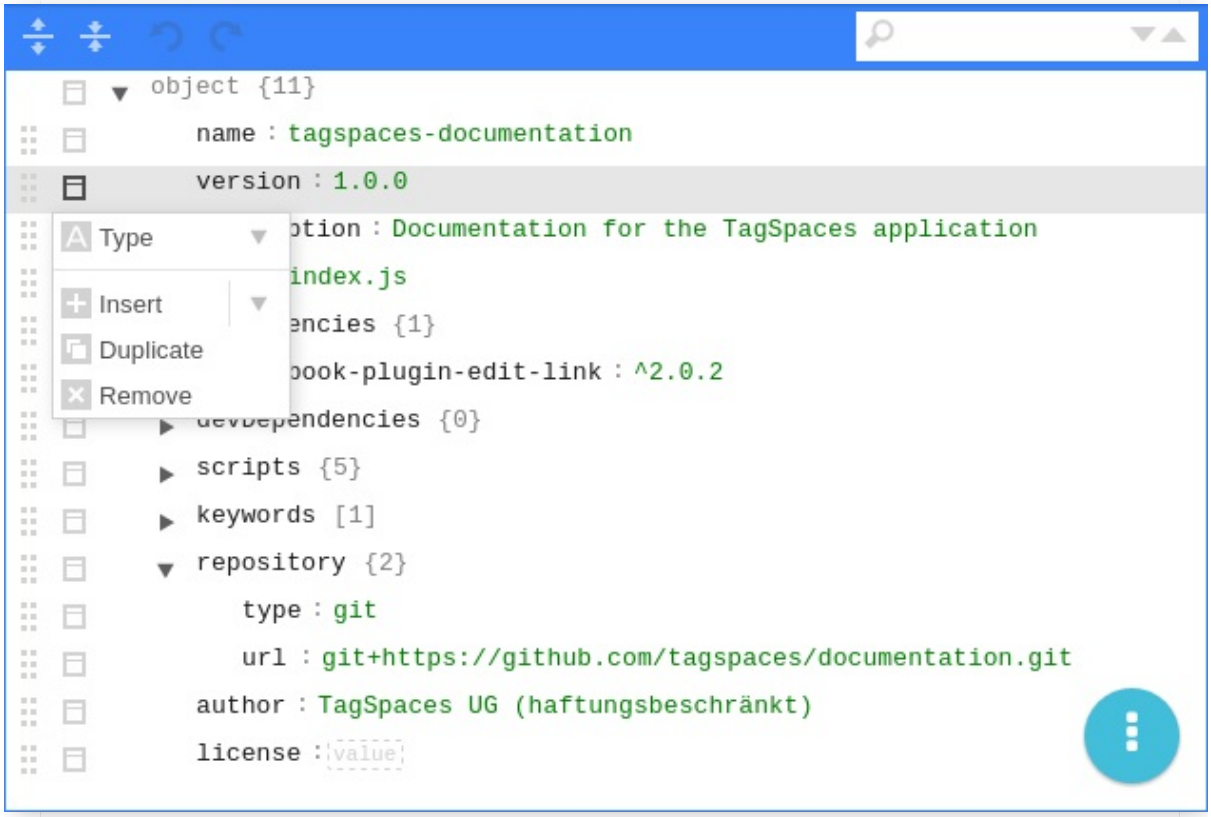
For example he link `[link text](images/picture.jpg)` will show the file named `picture.jpg` (located in the `images` subfolder of the folder your file is located) in the markdown viewer; while `[link text](files/more_files/example.pdf)` will open the file named `example.pdf` (located in the `more_files` subfolder of the `files` folder, located in the current folder) in an external PDF viewer.

To enter a relative path, you can usually use UNIX style slashes (`/`) in paths, e.g. `path/to/file/filename.ext` , as `node.js` upon which TagSpaces is built, will handle them properly even on Windows. This allows for interoperability across different Operating Systems. If you only use Windows however, you can use a backslash (`\`), e.g. `path\to\file\filename.ext` , if you prefer to, but such paths will not be understood on any other system, including Android.

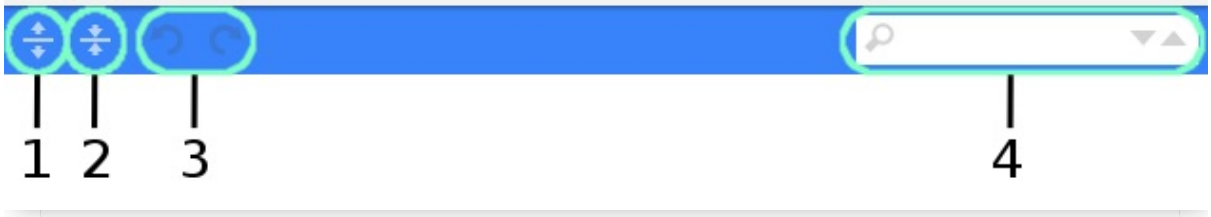
> **Hint** TagSpaces only understands relative paths. You cannot reference any level above your current folder, or the root of your connected location, but might only link files located in the currently active folder, or an subfolders within.

JSON Editor

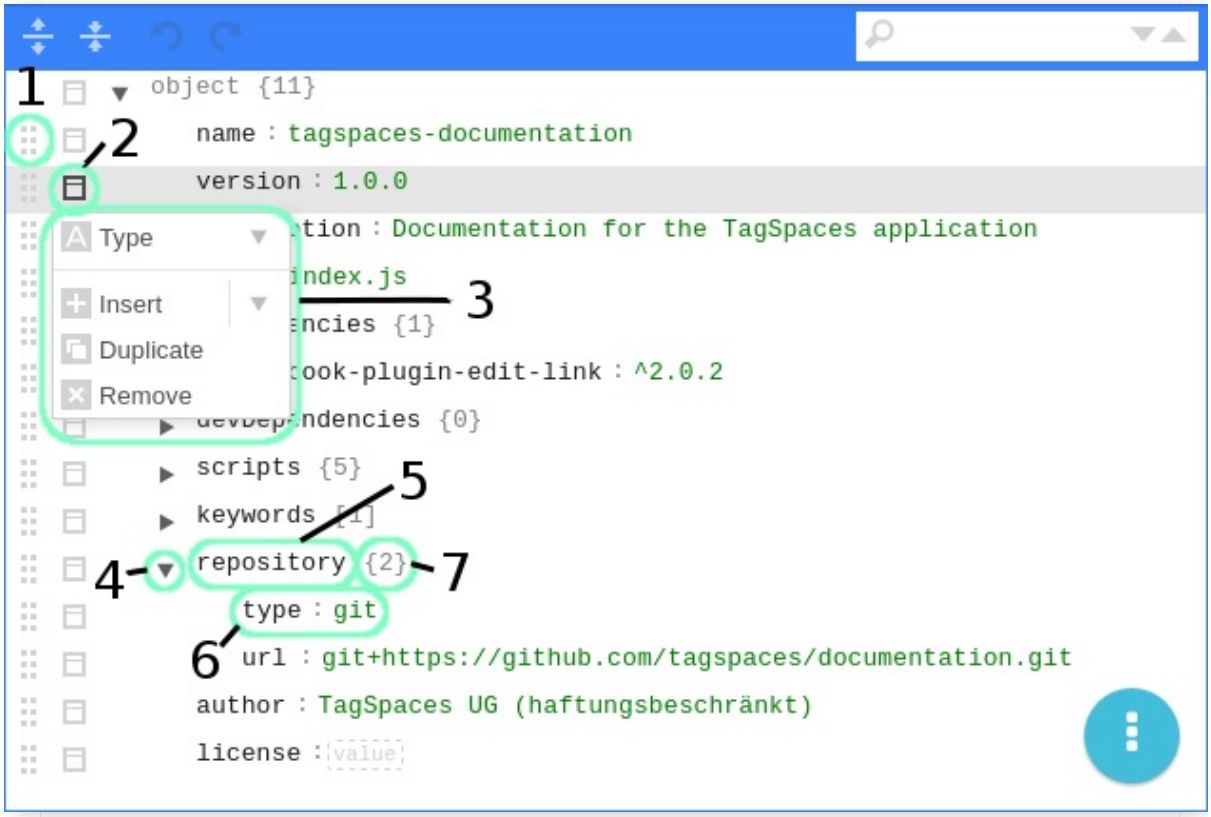
Editing `JSON` files will build upon the functionality you've already seen in the **JSON Preview mode**.



The top row of the editor will offer a basic toolbar, with some common actions such as **Expand all fields** (**1**), **Collapse all fields** (**2**), **Undo** and **Redo** (**3**) and a search box (**4**)



Each field has some useful controls that can help manipulate and rearrange `JSON` files easily.



- On the left edge, there is a drag handle, allowing for easy movement of each row. (**1**)
- Next to the drag handle there is an **Action Menu Button** (**2**), which opens the **Action Menu** (**3**), offering useful actions on each field, such as:
  - You can choose or change the **Type** of the field, choosing from *Auto*, *Object*, *Array*, or *String*
  - You can **Insert** an *Array*, *Object*, or *String* inside any field. (There is also an *Auto* mode for insertion). Inserting will place the new field **before** the selected field.
  - When selecting the **Action menu** on the last member of an *Object* or *Array*, you can also **Append** the same categories, which will place the new field **after** the last item.
  - *Objects* and *Arrays* will also allow for **Sorting**, as either *Ascending or* Descending*
  - Finally you can **Duplicate** or **Remove** any field from the hierarchy, (apart from the root)
- To the right of the Action menu button, there is an **Open/Close chevron** (**4**), which can expand or collapse each field.
- Finally you have the field itself, with one (**5**) (for arrays and objects) or two editable sections (**6**) (for string type), and an optional, read only information field (**7**), that displays the number of sub-fields, in either curly brackets `{ }` (for objects), or square brackets `[ ]` () for arrays.

The **FAB Overflow Menu** will offer to either **Print** the file, or display a **JSON Help** menu, which gives a summary of the numerous key combinations that can be used to work on JSON files even faster.

Organizing your files and folders with tags

Motivation

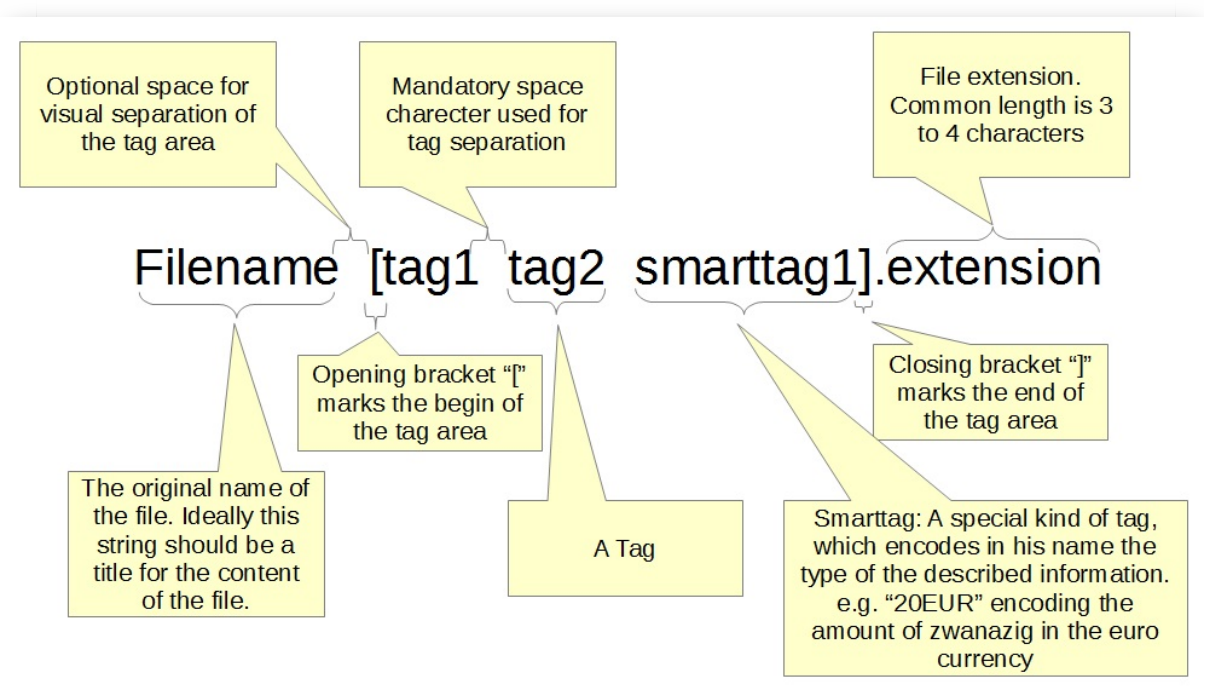
Tagging and tags are a fresh approach to categorizing and grouping things. Tagging allows the user to label thing with their own words. They don't need to conform to keywords or categories created by somebody else. Tagging allows you to describe what you're seeing in your own words, to imbue the experience with your own meaning. Tagging saves your time, and can makes searching for a specific song, movie, book, document, note, or whatever you're looking for, much easier and faster.

Tags are personal things. Tagging something is your emotional response and not simply a taxonomic decision. With using tags, you have the control to define things for yourself, and on the web -- not only for music, literature, games and movies, but also public websites, data repositories and consultations. Tagging is a new indirect way of control to express your mind and to name things your way -- you can call it a form of "freedom of speech", if you want to go that far. Tagging is a power to create your own genres in movies, music popular culture, or basically anywhere.

Key functionality in TagSpaces is the ability to add tags to files and folders. In comparison to similar product, **TagSpaces does not use a central database for storing the tags** but rather offers to alternative ways for saving this meta information, which are described in the following sections.

Tagging based on filenames

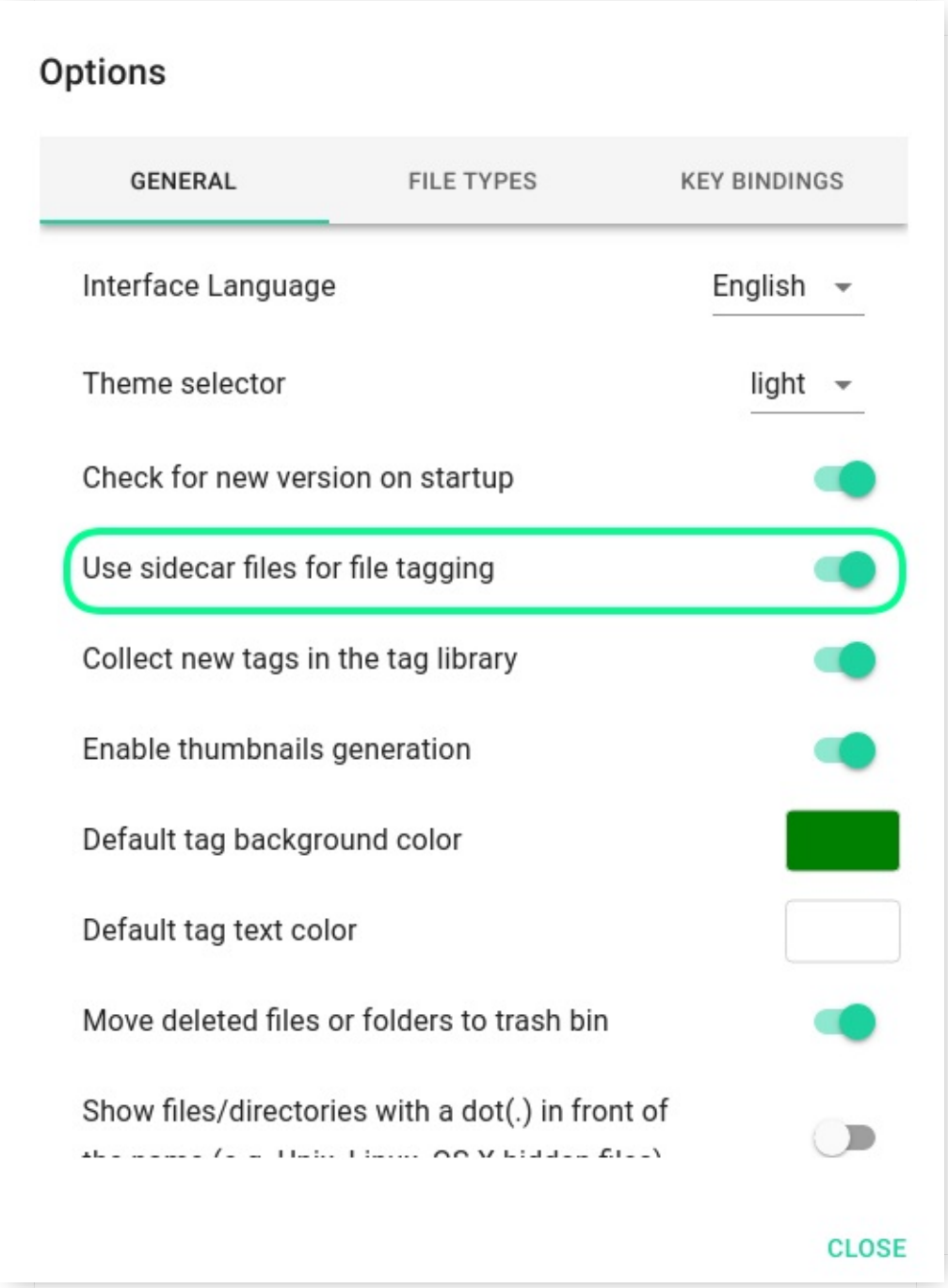
TagSpaces supports tagging of files in a cross platform way. It uses basically the name of the file to save this kind of meta information. As an example if you want to add the tags `vacation` and `alps` to a image named `IMG-2653.jpg`, the application will simply rename it to `IMG-2653[vacation alps].jpg`. File renaming is of course very controversial solution, with its own limitations (on some operating systems the file path length is limited to ca. 256 characters). Once embedded in the name of file, the tag stick there and can be removed only by file renaming. **This makes the tagging "durable" and portable**. The tags embedded in the name of a file "survives" synchronization across cloud platforms such as Dropbox and Google Drive and can be read by TagSpaces or any other file searching software on Windows, Mac OS, Linux or Android.



Tagging with sidecar files

As alternative to saving the tags in the file names, TagSpaces offers saving this kind of data in a sidecar files located in a hidden `.ts` folder. This method is used for adding tags to folders and can be used for adding tags to files. The activation of this kind tagging for files, can be done in the settings of the application as shown in the following screenshot.

> **Note:** Please note that by default the `.ts` folder is hidden only on OSX and Linux operating systems, on Windows folders with dot in front of their name are not hidden by default. Starting with version 3.0, TagSpaces makes hidden the `.ts` folder under Windows automatically.



Activating the persisting of tags for files in sidecar files

After the activation, the application will created for every tagged file an extra file having the same file name as the source file, but with a additional JSON extension. For example after tagging some files in some of your file locations you will have a similar file structure.

```
~ location (with your files)
├── subfolder
│   ├── .ts
│   │   ├── file1.jpg.json
│   │   └── file2.pdf.json
│   ├── file1.jpg
│   └── file2.pdf
├── .ts
│   ├── file3.png.json
│   └── file4.docx.json
├── file3.png
```

```
└── file4.docx
```

The main advantage of this solution is that the name of the files is not changed after tagging and there is theoretically no limit in the number of tags you can add to a given file. If you move or rename tagged files in TagSpaces it will take care of the sidecar file, which will also renamed or moved in the appropriate folder. But if you move or rename this file in an external file manager, you have to move or rename the sidecar files by hand. The same hold true for the deleting a file from an external application. It will not automatically delete the sidecar file in the `.ts` folder. These drawbacks makes the tagging with sidecars less robust and future proof.

> **Note:** If you want to have the files located in the `.ts` folder synched with some cloud service such as Dropbox or Google Drive you have to enable the synching of hidden folders and files.

Timestamp as default tag

When you create a new file in TagSpaces, a time-stamp in the format of YYYMMDD~hhmmss will be automatically added as a tag, where

- `YYYY` means the current year
- `MM` - the current month
- `DD` - the day
- `hh` - the hour
- `mm` - the minute
- `ss` - the second

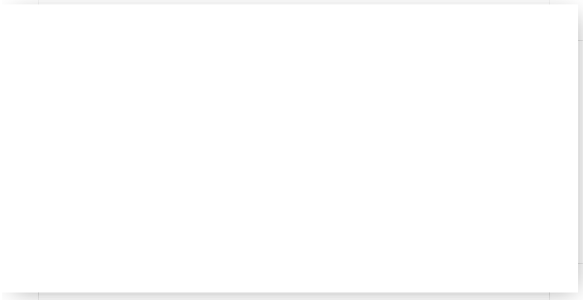
when the file was created. For example a file created on the 17th of January in 2017, at 10:30 (and 32 seconds) would be tagged with timestamp like `20170117-133032`

Such timestamps make it possible to easily group files from different locations based on the time of creation. Of course this would be possible by reading the same information from the file system, but tagging with the timestamp allows for greater flexibility. For example you can change the timestamp to anything you like, while keeping the original creation modification date of the file, and tagging the file will allow you to use TagSpaces's powerful grouping and organizing features in the same place you manage other tags.

Tagging with Drag and Drop

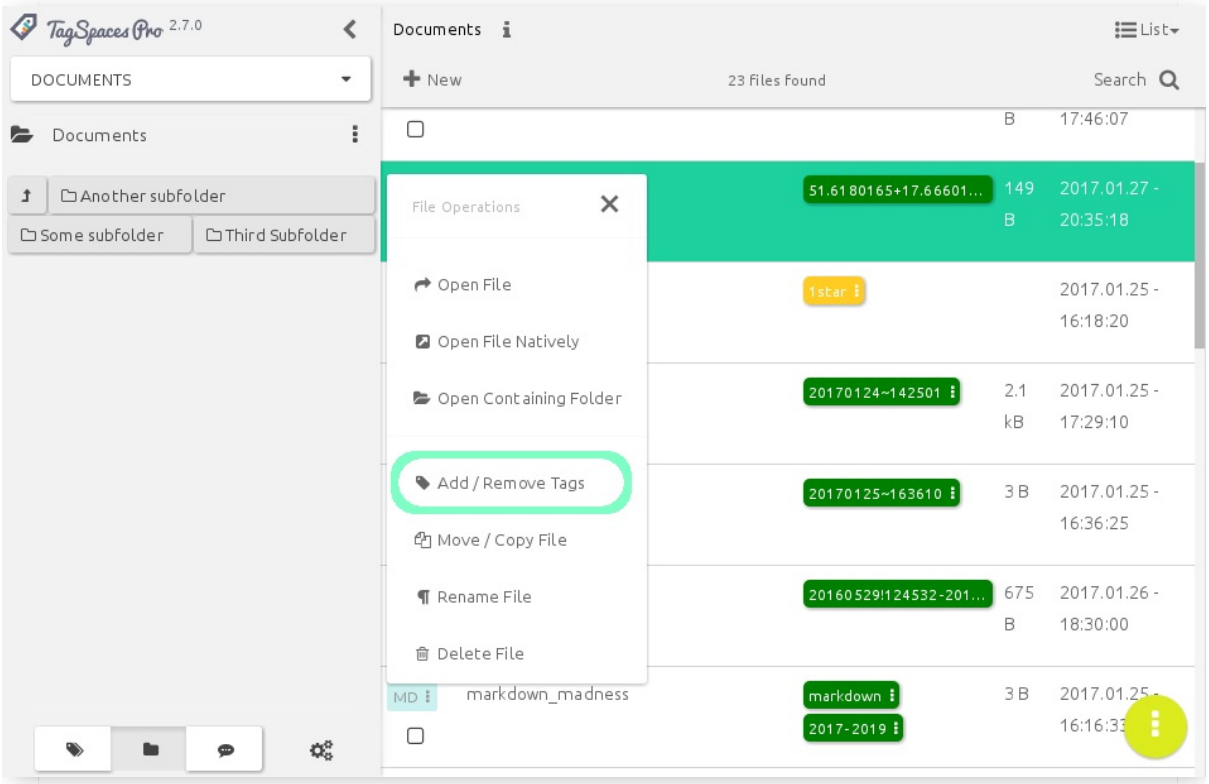
The user interface was designed for touch screens, but it also supports drag & drop operation for the desktop, such as:

- Adding tags from the tag groups to files in the List and Grid perspectives and to the file viewer
- Moving tags from one tag group to another

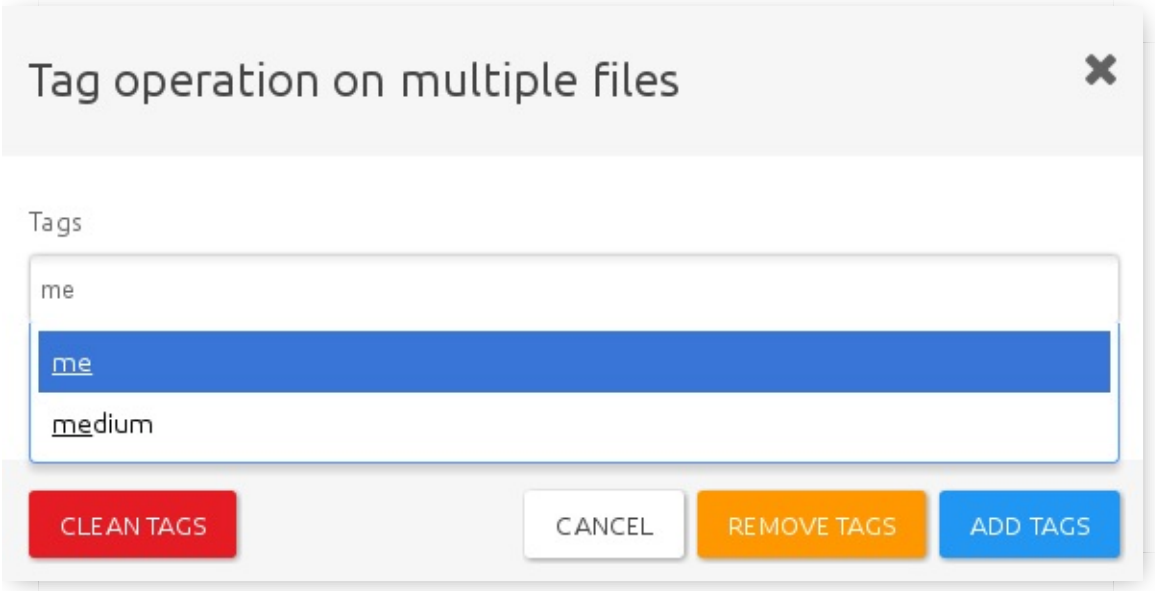


Tagging using context menus

A context menu can be accessed by either clicking on the file extension icon of a single file (in List Perspective), or by right clicking a file or a selection of multiple files. From the context menu, select *Add/Remove Tags*



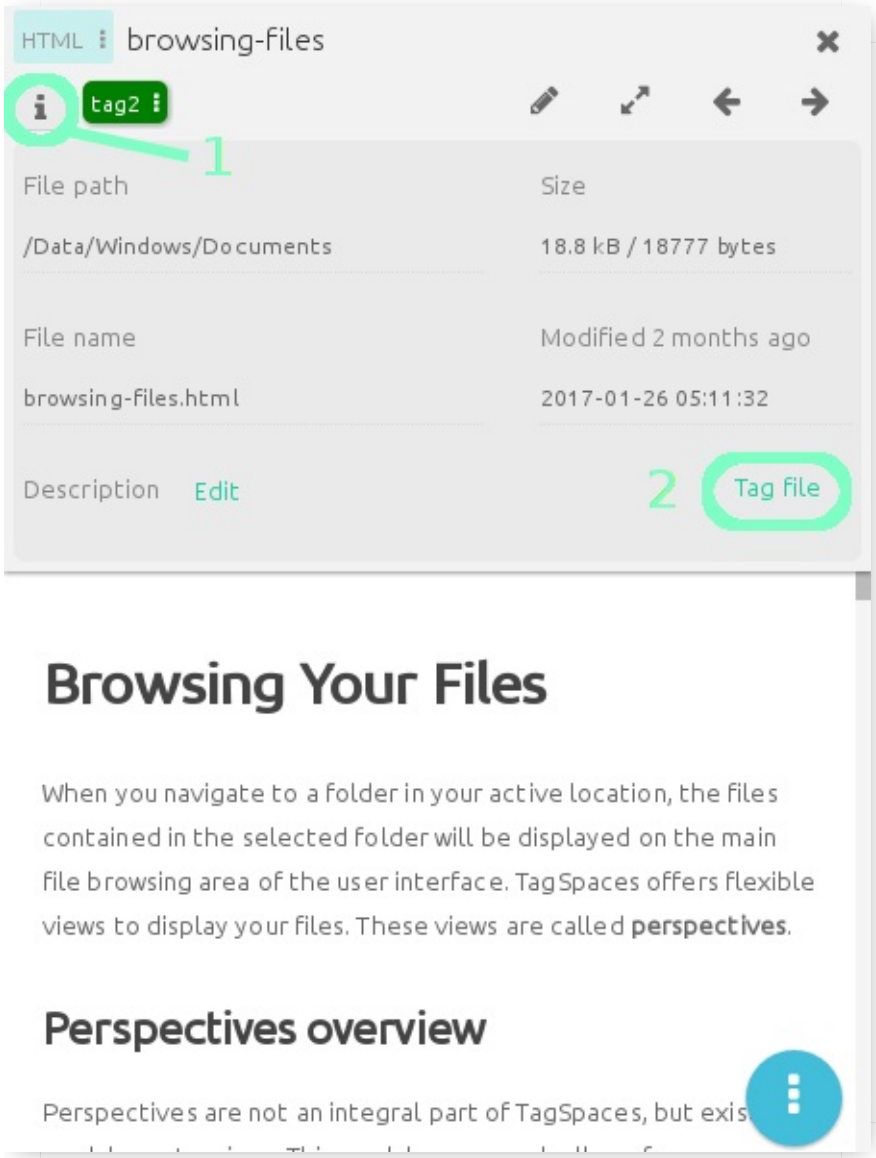
to access a popup dialog, that allows you to manage tags on the file. This popup dialog will not display the currently active tags, but allows you to specify tags by name, offering suggestions based on tags currently in the tag library.



The options you have here are:

- **Clean tags**, which will remove all tag information form the selected files
- **Remove tags** will remove the specified tags from the files
- **Add tags** will add the specified tags to the selected files

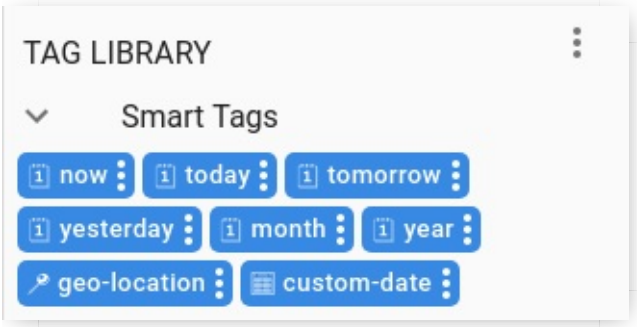
The popup dialogue can also be accessed from the file preview area, by clicking the file info icon (**1**), and selecting *Tag File* (**2**)



Smart tags

Smart tags are one of the advanced tagging features of TagSpaces. Smart tags can be either **date and time**, or **location based**, and offer convenient **dynamic tagging** tagging, based on a range of criteria.

- **Time and date tags** are timestamps textual representations, such as *now*, *today*, *tomorrow*, etc. Applying one of these tags to a file will add a timestamp, corresponding to your choice of smart tag. A timestamp has the format of `YYYMMDD-hhmmss`. Tagging a file with e.g. *now* would apply the full timestamp down to the second, while tagging e.g. *month* would tag it with a subset like `YYYYMM`. Smart tags have a distinguishing blue background, which they only retain in the tag library.
- pro Applying a **geo tag**, an additional feature in TagSpaces PRO, opens up a dialog with an interactive map, where you can drop a pin. The geo-location (longitude and latitude coordinates), will be added to the selected file as a tag.



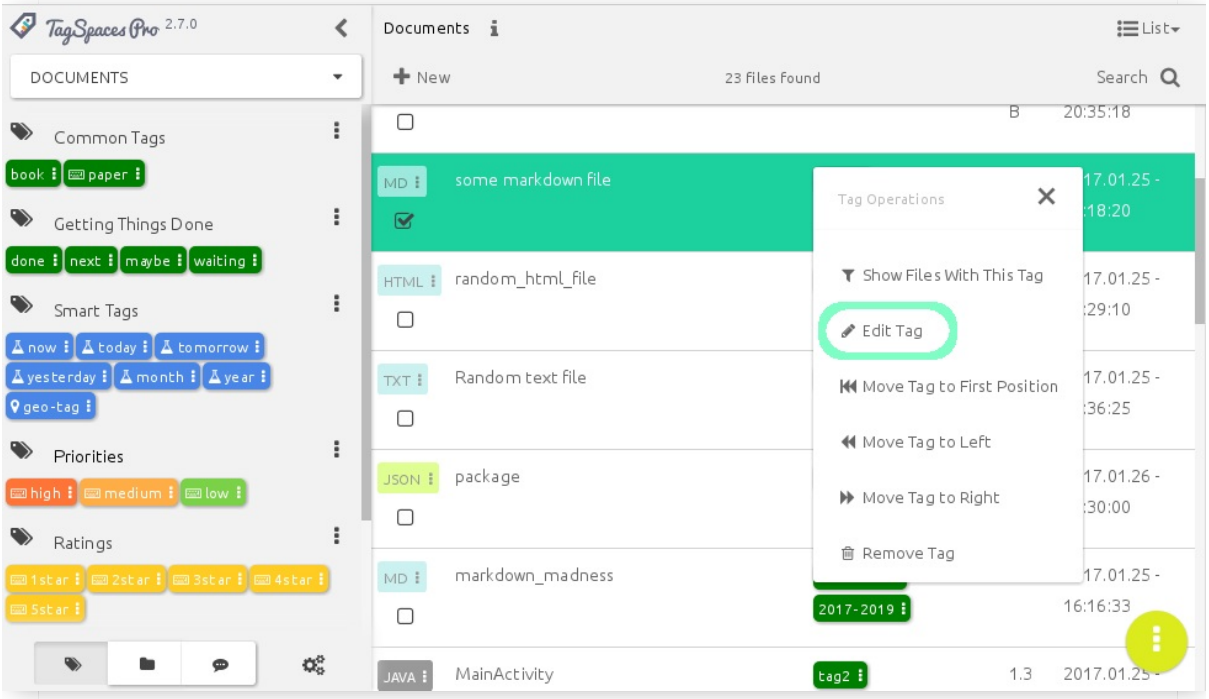
Date and time based smart tags

These smart tags offer a quick and easy way to timestamp documents or files, with different levels of precision. Dragging or applying a smart tag to a file will create a tag based on the current time date. Currently the following tags and formats are available:

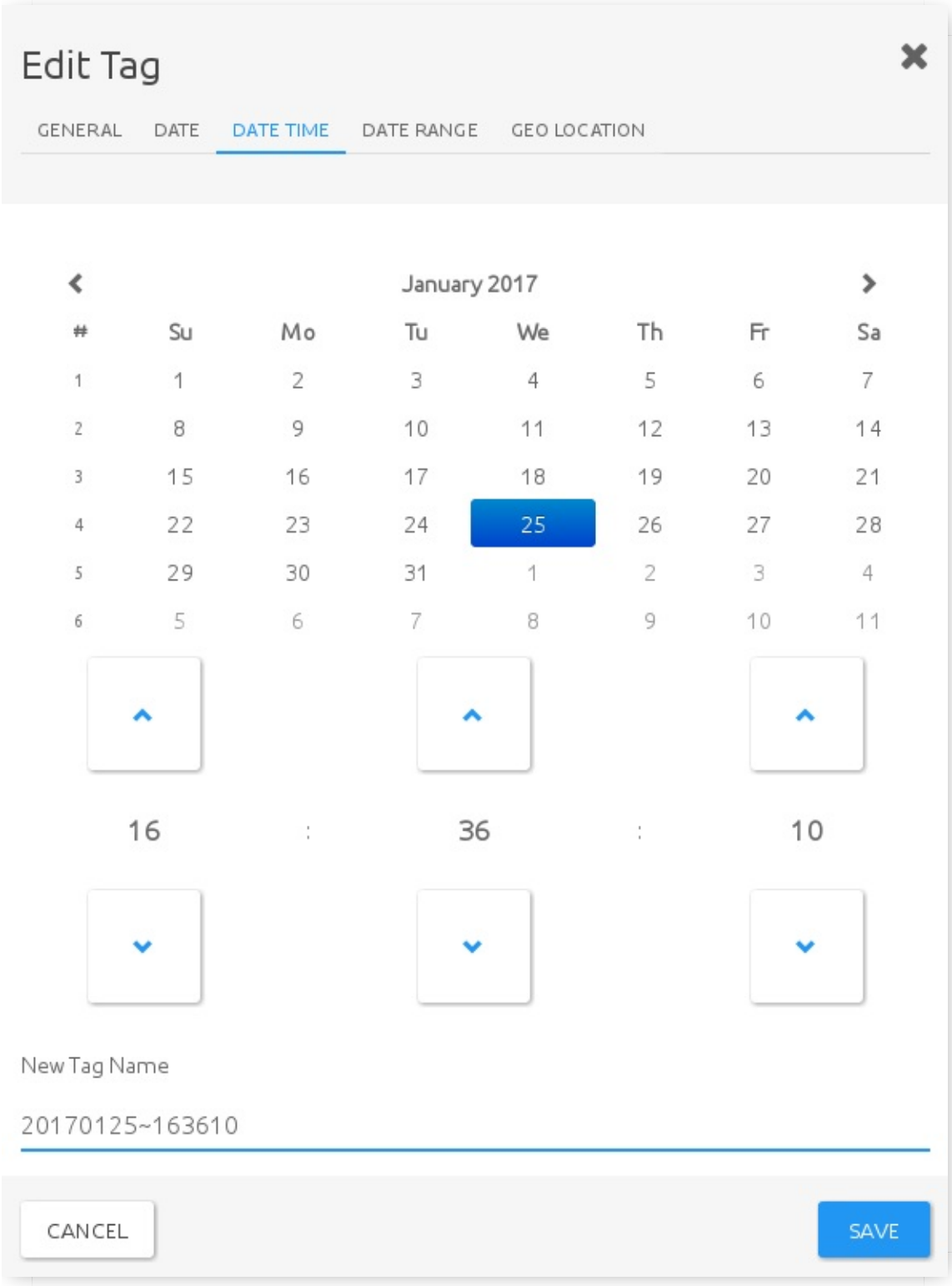
- **now** - This tag will create a very precise timestamp (from current year, down to seconds) of the moment you have applied it, in a format of `YYYMMDD-HHmmss` , e.g. `20170314-145021`
- **today**, **tomorrow and** yesterday** will apply a timestamp with the current, the next, or the previous day's date, in the format of `YYYYMMDD` , e.g. `20170314` .
- **month** will create a timestamp of the current month, in the format of `YYYYMM` , e.g. `201703`
- **year** only applies the current year, in the format of `YYYY` , e.g. `2017`

> **Note:** Every newly created file will automatically have a smart tag, equivalent to **now**.

Smart tags, once applied, can be edited, or further refined be refined to include date ranges. Clicking on a tag, ans selecting *Edit Tag* form the context menu



will bring up a popup dialogue also called **Edit Tag**

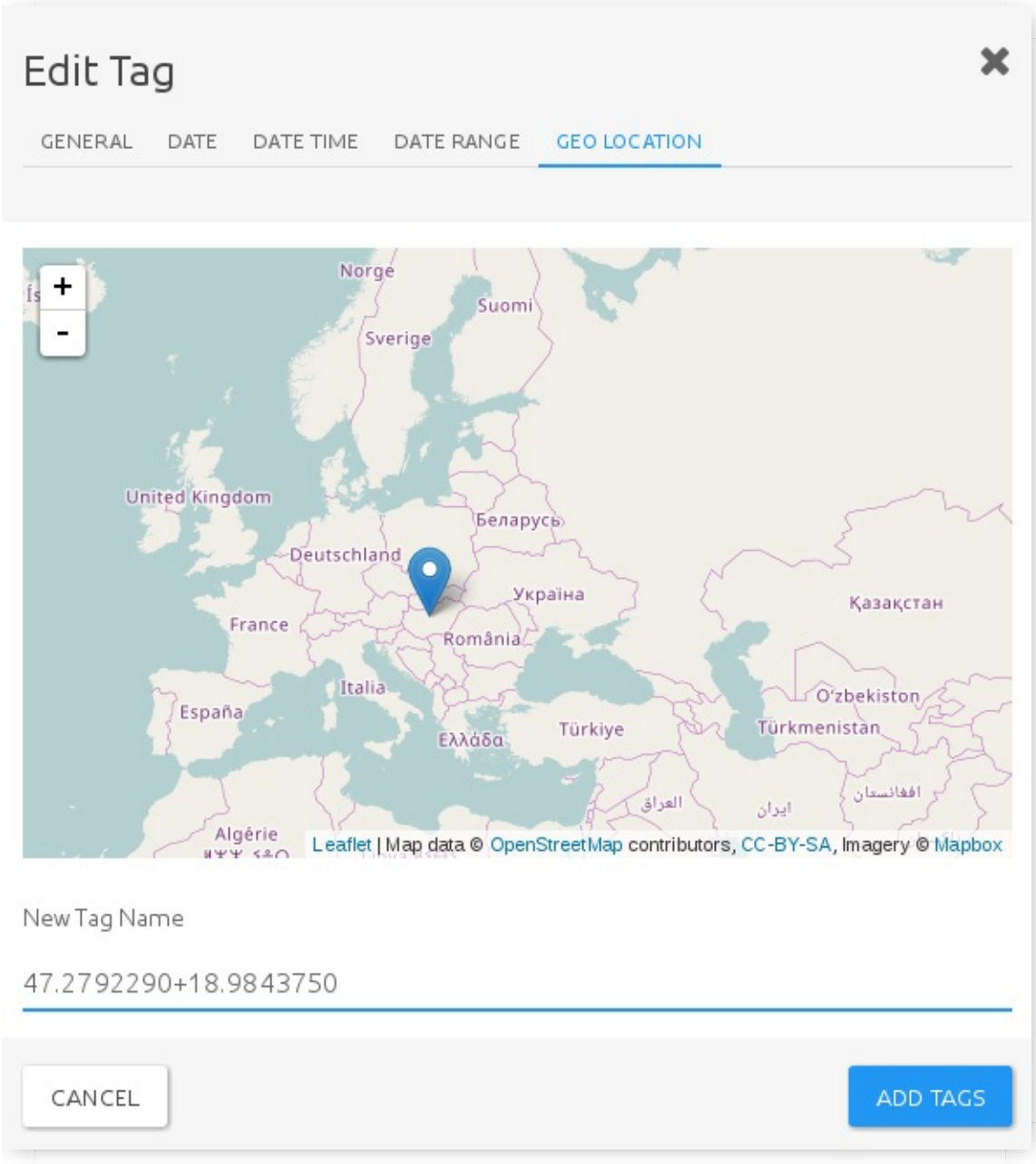


besides being able to easily modify date, or date/time tags on their respective tabs, you can also specify date ranges in the following formats:

- **Year Ranges:** 2016-2018
- **Month Ranges:** 201605-201701
- **Date Ranges:** 20160531-20160603
- **DateTime Ranges**: 20160529~124532-20160529~154500

pro Geo location tagging

The geo tag is a special kind of smart tag, available in TaSpaces **PRO**, that allows tagging files with precise geolocation coordinates. When You drag the tag "geo" on a file, the **Edit tag** popup window with the Geo Location tab will open, showing a map from OpenStreetmap. You can drop a pin anywhere, and move it around, it automatically becomes a tag in the format of `latitude+longitude`, e.g. `47.2792290+18.9843750` This tag will then be applied to the file and treated as a geo-smarttag by TagSpaces



Priorities and ratings

These special tags are useful for organising files by either importance or quality. You can apply priorities `high`, `medium` and `low`, and start ratings from `1start` to `5star`. Star ratings are yellow by default, whereas priorities are colour coded to easily distinguish visually.

Users can easily extend these tags: You can simply create a new tag and add it to e.g. the priority group, give it a name a colour and a key binding, and you are all set.

Searching for files and folders

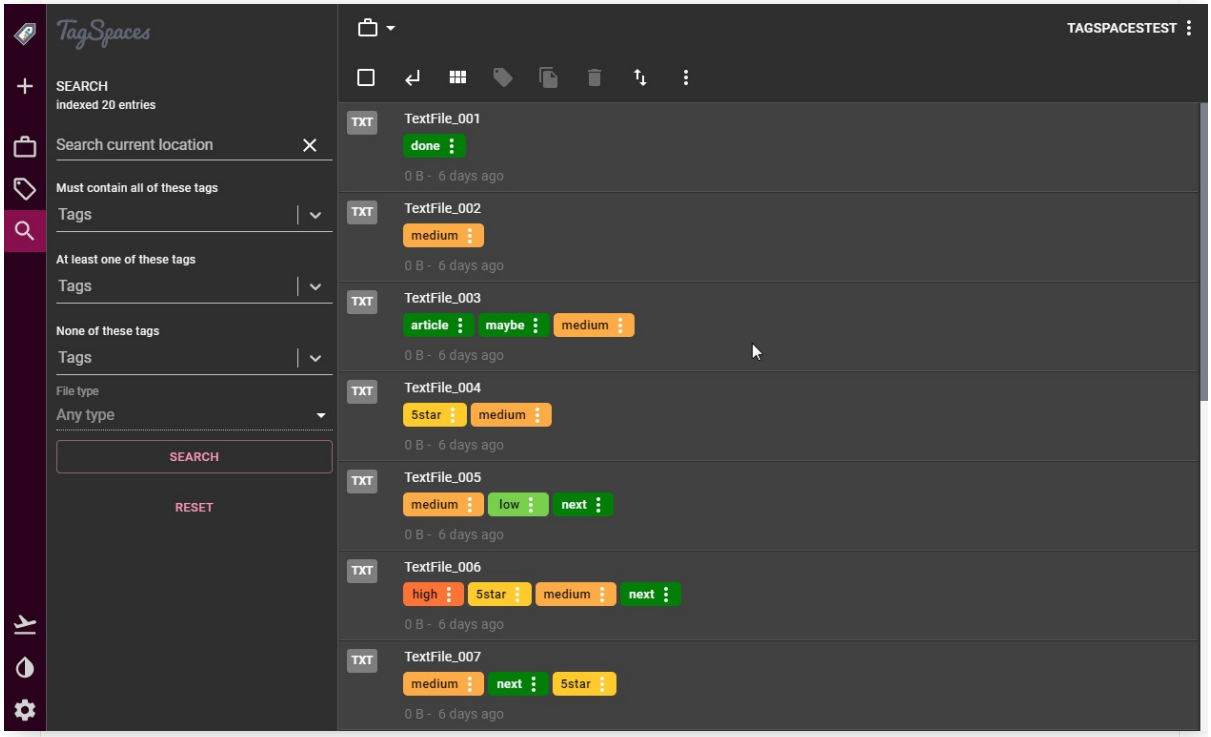
TagSpaces has an integrated file and folder search functionality. The difference to the classic Desktop Search software is that the search is performed only in the currently opened location or folder. The search is based on a **index**, which is created immediately after the user opens a given location and has the following functionality:

- The search query in the free version consists of two components. The first one is just a simple free text which searched in the index. The second component is a list of tags. At least one of the tags should be assigned to a given file or folder in order to be included in the search results.
- The search algorithm considers with different weight the following fields from the index.
  - The file or folder name
  - The tags assigned to the file or folder
  - The description added to the file or the folder
  - The file or folder path
  - The content of txt, md and html, if the full text search is activated for the current location (TagSpaces PRO only)
- The search algorithm has a build-in fuzziness, meaning that if the free text search query contains a misspelled word, TagSpaces will try to find files and folder with the best ques.
- The tag search is strict, without fuzziness, if you are not sure about the exact name of a tag, write in the free text query.
- The search is case insensitive.
- The user can switch to the search area by clicking the Ctrl+3 / Cmd+3 key combination (configurable in the Setting).

Basic search

In order to provide a boolean search support for tags, the search user interface for tags was split in three input fields:

- Must contain all of the tags - all of the tags listed here should be attached to the files or directories (**logical AND search**)
- At least one tag - any file or folder which contains one of the specified here tags will be included (**logical OR search**)
- None of these tags - entries which have one the tags listed here will be excluded from the search results (**negative search**)



Short video showing searching for tags in action

> **Note:** After opening of a certain location, the application starts to index in background all the files and folders from this location recursively. During the indexing time the search is not available to the user, but all other functionalities are accessible. Opening of location containing more the 100000 files could lead to performance issues, during the index or later by search.

pro Advanced search

In addition to the searching by files name and by tag, the search in the PRO versions supports the following features on top:

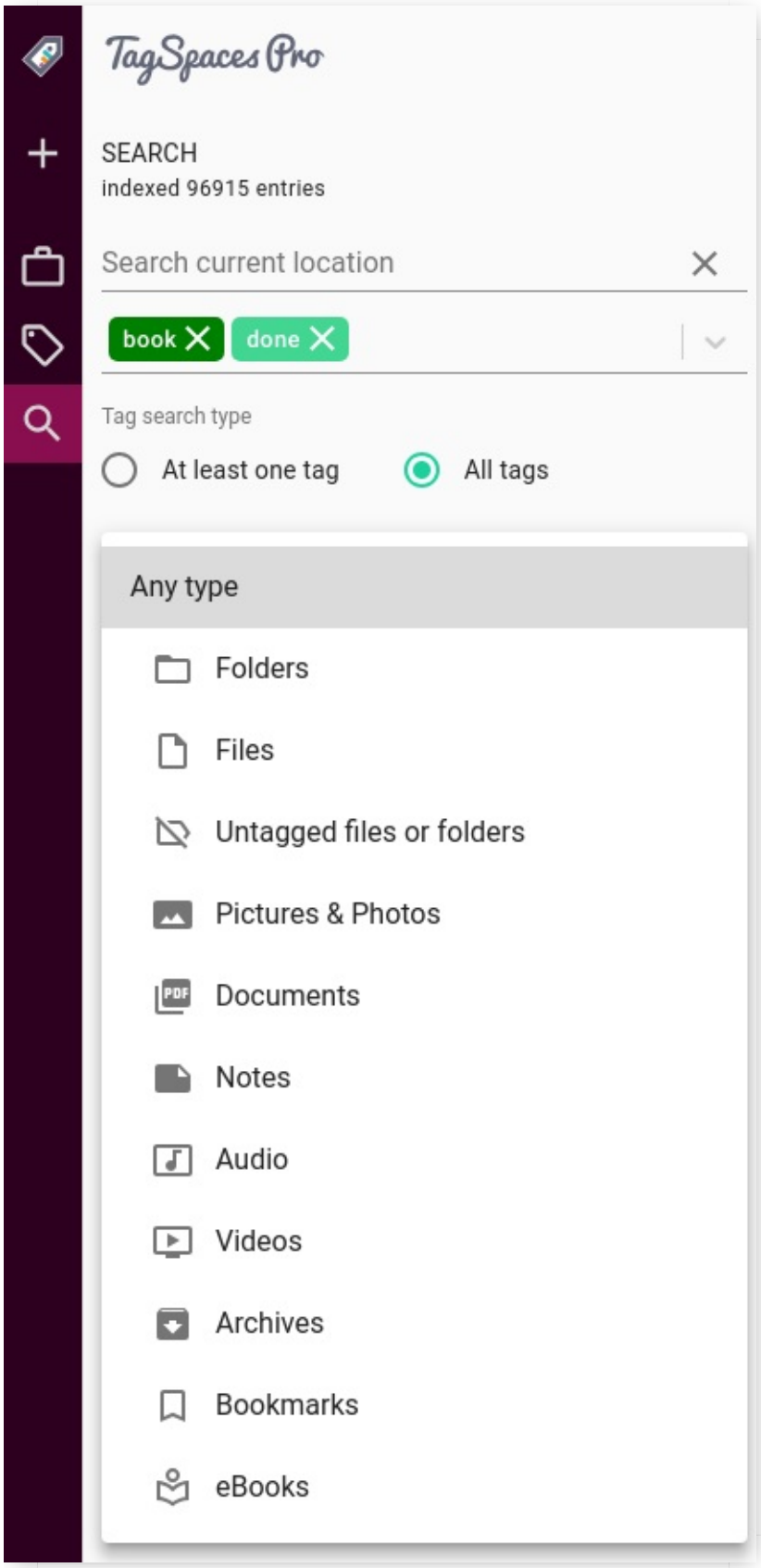
pro Search by type

In the file type dropdown you can specify types of the files you want to search. The following groups are supported:

- Pictures and Photos: JPG, PNG, GIF, etc.
- Documents: PDF, ODF, DOCX, EXL, etc.
- Notes: MD, TXT, HTML, etc.
- Audio files: OGG, MP3, WAV, etc.
- Video files: WEBM, OGV, MP4, etc.
- Archives: ZIP, RAR, TGZ, 7Z, etc.
- Bookmarks: URL, LNK, etc.
- eBook EPUB, MOBI, AZW, PRC, etc.

In addition to that we offer some special filters:

- Folders - showing only folder
- Files - showing only files
- Untagged files or folders - showing only files and folders witch are not tagged



Screenshot of the advanced search options

pro Search by file size

pro **Search by last modified date**

pro **Search by time period**

pro **Search by GPS coordinates**

Web clipping - collect everything online

Introduction

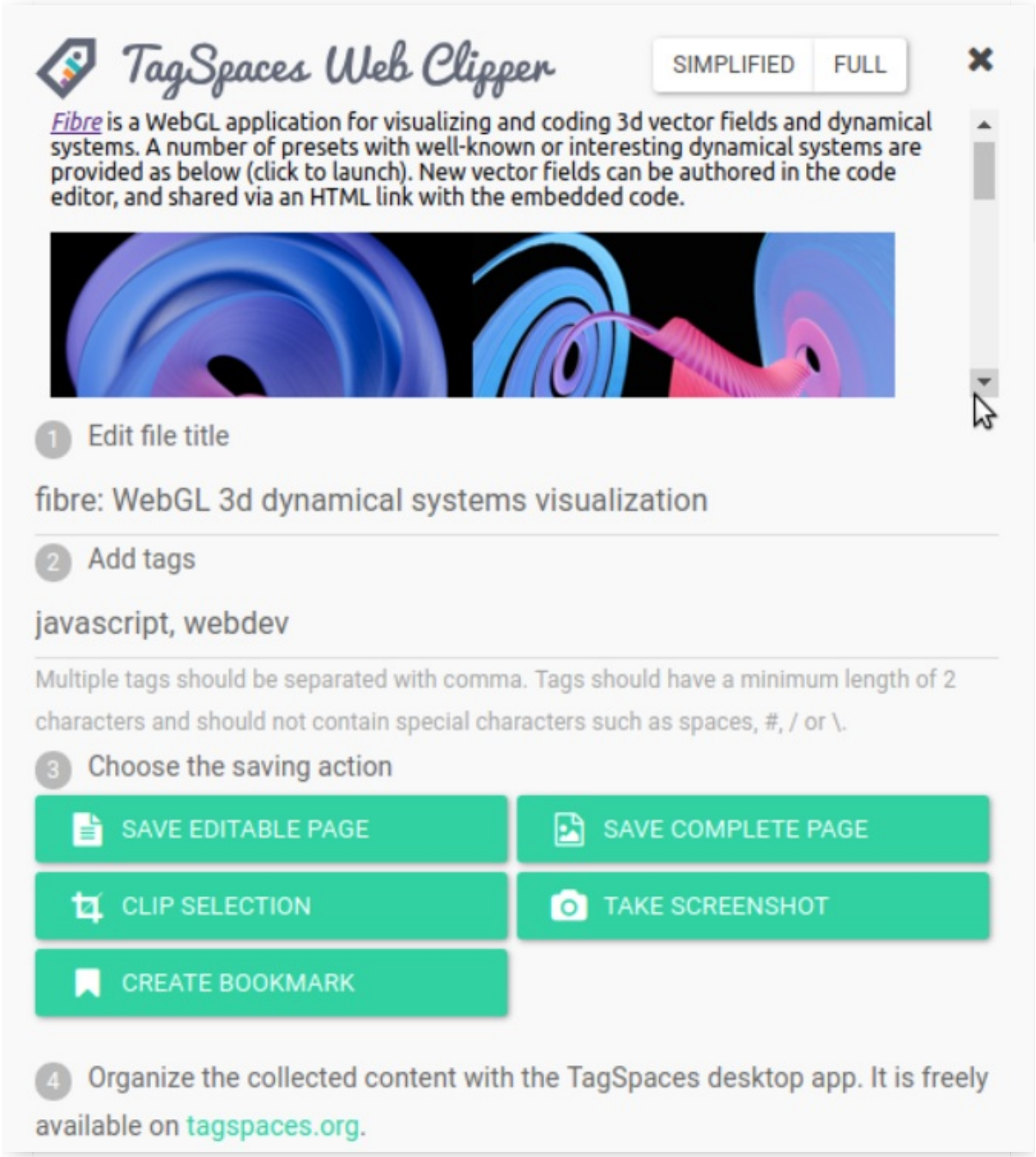
In order to provide an easy way to collect web content such as web pages, articles, PDF-documents, bookmarks, places and screenshots, we have created the TagSpaces Web Clipper browser extension.

Basic features

The browser extensions are available for Chrome and Firefox. Both extensions have the following functionalities:

- Saving the current webpage as a single file including the embedded images and styling information in HTML format. Here the extension supports two modes. The default one is called **simplified**, where TagSpaces uses a library for automatic extraction of the webpage's main content without any clutter of adds or navigation. This is very useful clipping articles for example. The second one is called **full**. Here the extension tries to save all the original text and image content of the webpage.
- On Chrome we support an additional file format called MHTML, which is preserving the original look and feel of the web page as much as possible.
- Saving the a selected part of the current webpages as HTML file. TagSpaces tries to embed the contained images as data-urls in the HTML file.
- Saving a screenshot of the visible area of the current web page as a PNG file.
- Saving an URL file containing the url of the current web page. This is useful if you don't want to save the whole page, but only to make a bookmark to it.
- Saving currently opened PDF-document locally.

Before the creation of any file, the user has the ability to change the title of file and to add tags to its file name.



A screenshot showing the web clipper in action

The basic functionalities are completely decoupled from the desktop application of TagSpaces and so they can be used with any other application supporting HTML, MHTML, PNG, PDF or URL files.

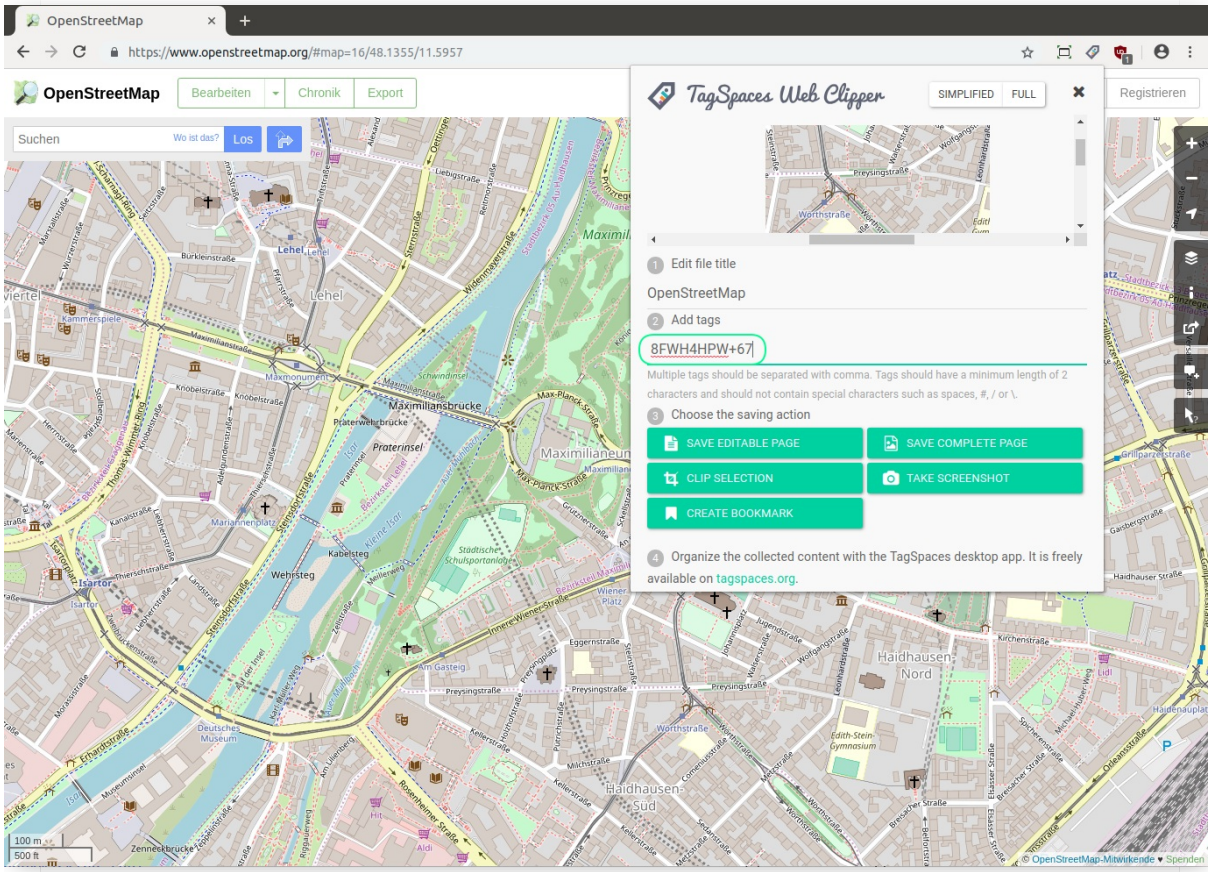
Advanced features

In addition to that we offers some features for more advanced use cases such as the following:

- Embedding the clipping timestamp and the source URL of the currently scraped web page in the HTML file. This information can be used later by previewing the file in TagSpaces for navigation to the original URL of the clipped page.
- Integration of a screenshot of the visible part of the web site in the created HTML and URL files. If you open the URL for example is opened in the desktop app, the screenshot is extracted and shown in the file preview area. It

is also used for the creation of the thumbnail for this file. In addition to that the screenshot is useful for archiving purposed, it displays the web page in the exact way you have opened it in the browser. Everybody knows that some page change or completely disappear very often. This feature makes TagSpaces a perfect visual bookmarking tool.

- Extracting the geo coordinates from the URLs of mapping services such as OpenStreetMap and Google Maps. This information is converted to a geo tag and embedded in the name of the created file.
- The extension can create the geo tag in Open Location Code or OLC for short used as plus codes in Google Maps for example. The plus codes have the advantage that they represent the geo coordinates in a much simpler and readable way.
- By saving of a screenshot from the current web page, the web clipper adds as tags the domain of this web page, the current date and tag "screenshot". This makes the search later for such screenshot much easier in TagSpaces and other application.



A screenshot showing the extracted geo location as plus code

The browser extensions are a practical additions to the desktop applications of TagSpaces, allowing a seamless way to collect locally and organize data from the web.

Chrome Fine-tuning

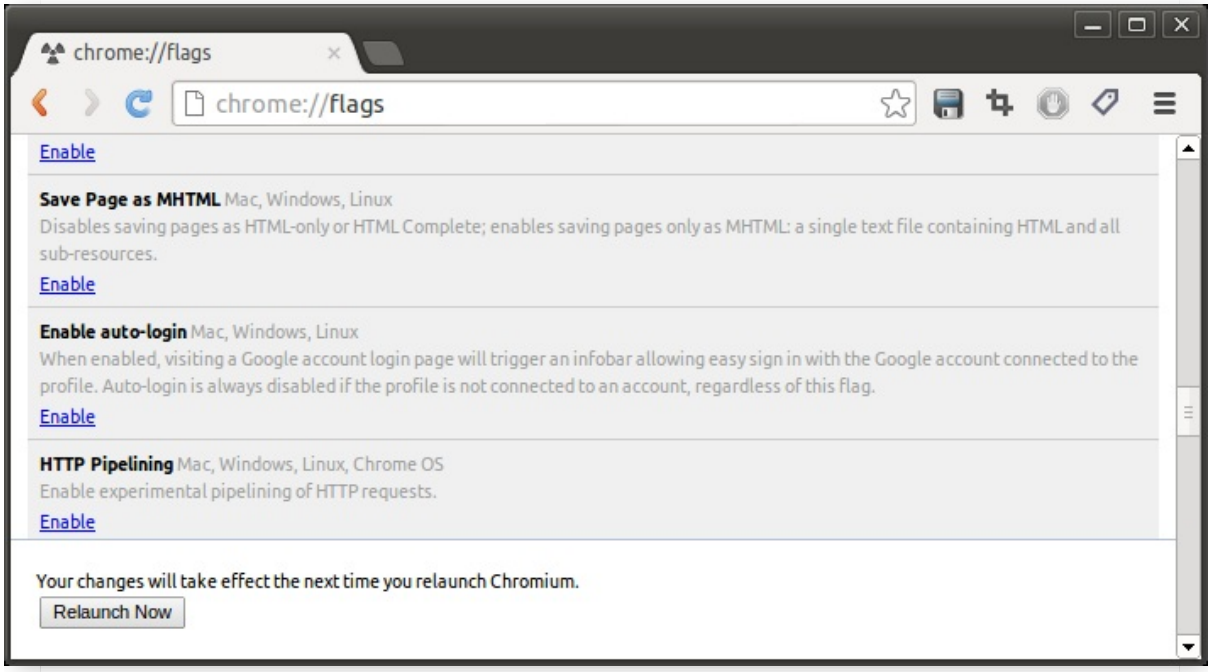
Here you will find some tips and trick for using the TagSpaces extension in the Chrome browser.

Enabling the saving of webpages as MHTML

TagSpaces is a great tool for MHTML file organization on many platforms, because it features an integrated MHTML viewer, but the question here is how you can save web pages as handy MHTML files directly out of the Chrome browser. Here you will find the answer of this question for the both browsers - Chrome and Chromium respectively.

And no, you don't have to install the TagSpaces chrome extension to achieve this, but just to execute the following steps:

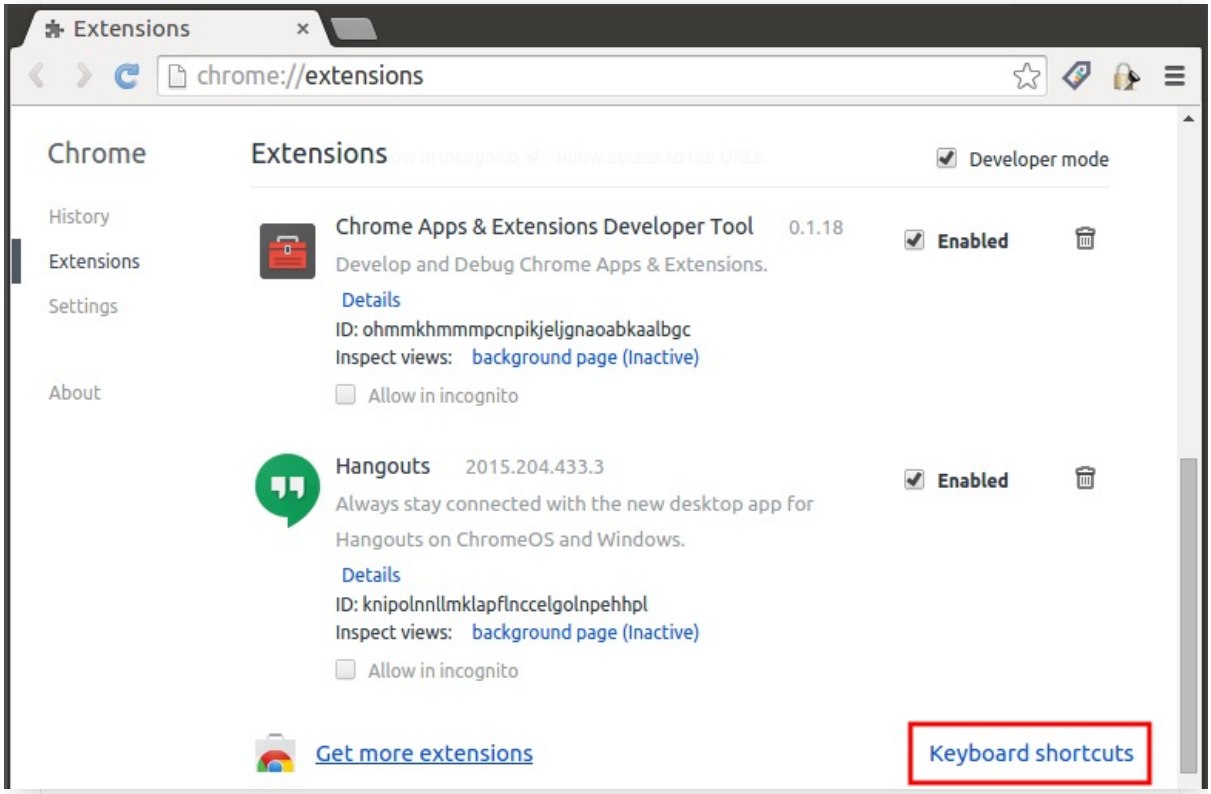
1. Start the Chrome/Chromium browser
2. Navigate to "chrome://flags"
3. Find the entry "Save Page as MHTML"
4. Click "enable"
5. Restart your browser
6. That's it, now the web pages will be saved by default as MHTML



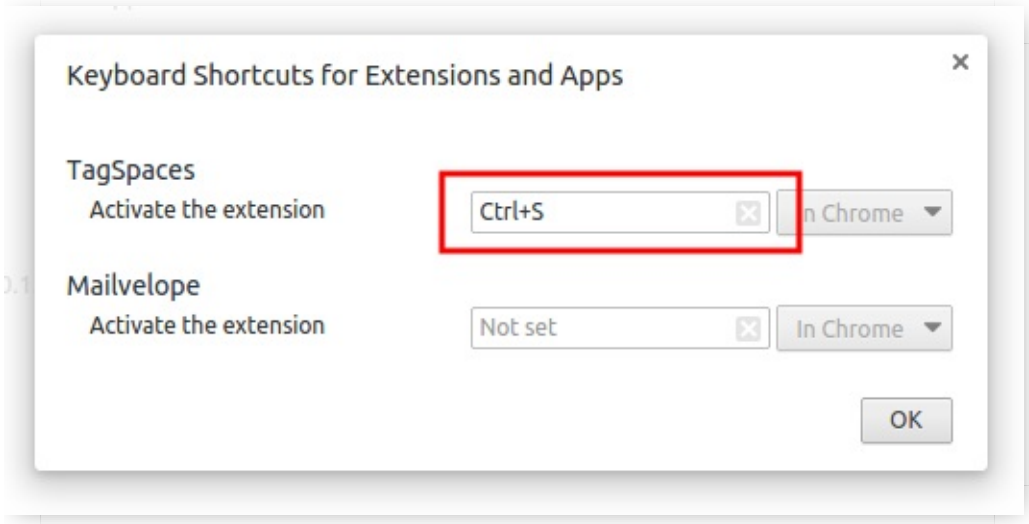
> **Note** After this activation you will not be able to save website in HTML anymore.

Adding keyboard shortcut to the web clipper in Chrome

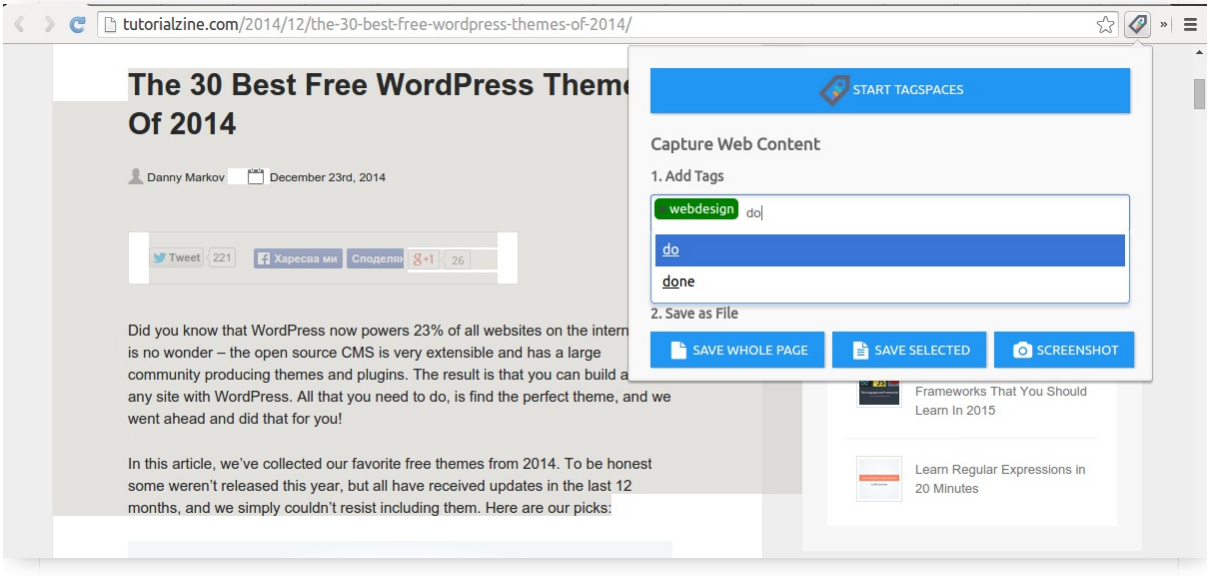
At the bottom of the extension management page in the Chrome browser you will find a link named "Keyboard shortcuts". See the red area of the screenshot below.



This link opens a dialog where you can set a direct keyboard shortcut, which will open the popup area of an extension. Since currently the main functionality of the this area in TagSpaces is to scrap the current webpage, I choose for myself the shortcut `ctrl+s`, which overwrites the default save as functionality of Chrome browser. You can choose of course any other key combination, like for example `ctrl+shift+s`.

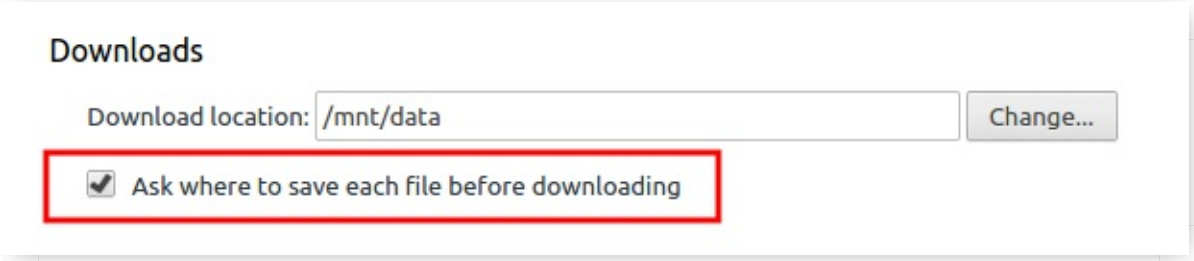


So now I can conveniently save and tags any page by just clicking this shortcut combination.



Download location for web clippings

In order to be asked every time, where you want to save the scaped web content, make sure to activate the checkbox "Ask where to save each file before downloading" in the advanced Chrome settings.



Syncing files between TagSpaces installations

Using cloudservices like:

- Dropbox (TM)
- Google Drive (TM)
- Microsoft One Drive (TM)

or other P2P projects like

- Syncthink (TM)
- Bittorentsync (TM)

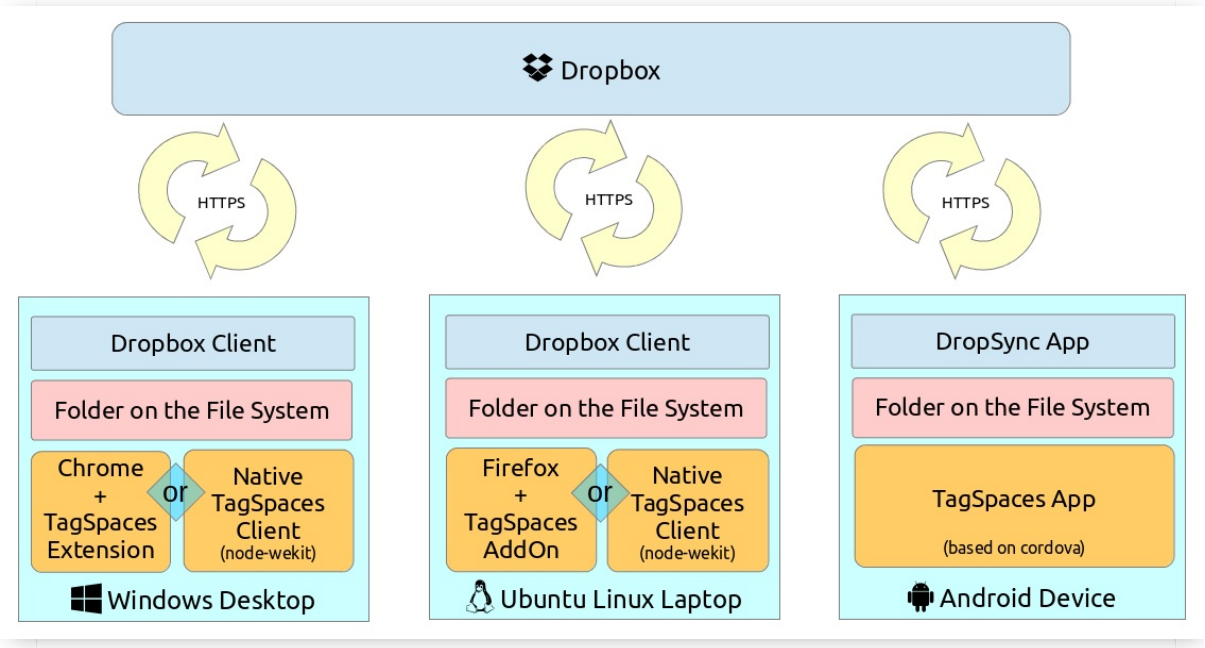
Encrypted:

- Cryptomator

One of the most common questions I receive about TagSpaces, is about the missing files synchronization feature. The concerns of the users are that they spend an hour tagging their files on the laptop, but now they want to get the same results also on their tablet or desktop computer and vice versa. Ok, my opinion here is that TagSpaces does not need sync functionality, because all the tags are saved in the names of the files, which makes this meta-information extremely portable between devices. This is one of the main differences between TagSpaces and other applications offering tagging on files. Most of those applications are using some kind of database to store the tags, which makes the transfer of this information challenging. Besides that, your tagging information in this case is locked by the vendor and you cannot migrate to another application or service without significant effort. Saving the tags in the file names make the information stick to your files, and you can find files based on the tags even with simple search functionality supplied with your operating system.

For the synchronization of the tagged files with TagSpaces, you can use any "cloud" service like Dropbox or projects like ownCloud, which provides sync clients. At present, I personally use Dropbox with its native clients on my Ubuntu laptop and the Dropsync app on my Nexus 10 and Nexus 5 devices. This tool combination works perfectly with TagSpaces and the way it preserves tagged information. And since there are TagSpaces clients for many platforms (currently Windows, OS X, Linux, Firefox, Chrome and Android), your tagged files can be synced and used almost everywhere.

The following diagram shows my current utilization of Dropbox in connection to TagSpaces.



If you are using TagSpaces with any other sync service, please post a comment bellow. I am always keen to know details on the way you are using TagSpaces.

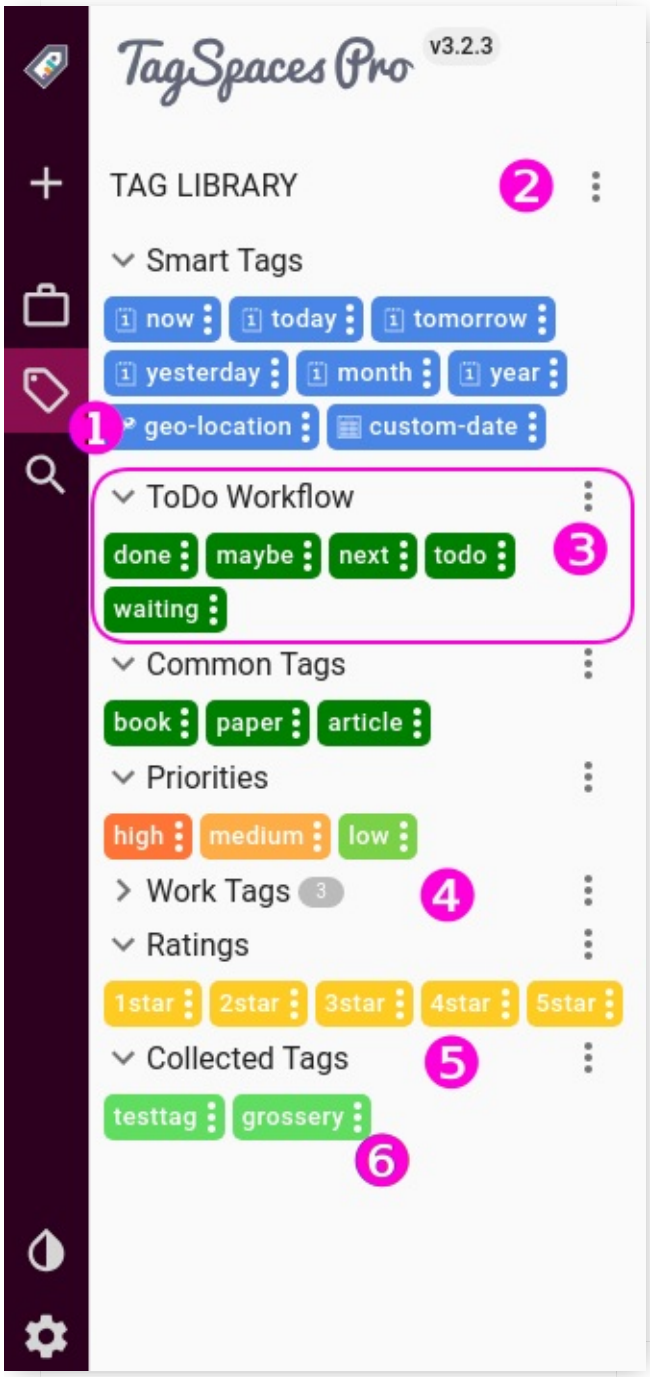
Collaboratively using tags and tag groups

Introduction

A common requirement for many TagSpaces users is sharing tag and tag groups across computer networks and devices. This article runs through the steps required for setting up TagSpaces in order to use a shared tag groups. It will guide you through the process of exporting the tag groups, editing them in externally and re-importing or reusing them in other installations. After understanding how TagSpaces handles tag groups you will be able to achieve the following use cases with our application:

- Sharing tag groups with other TagSpaces users on a shared network drive or shared Dropbox / Google Drive / Syncthing accounts.
- Using the same tag groups in other installation of the application on platforms such as Android

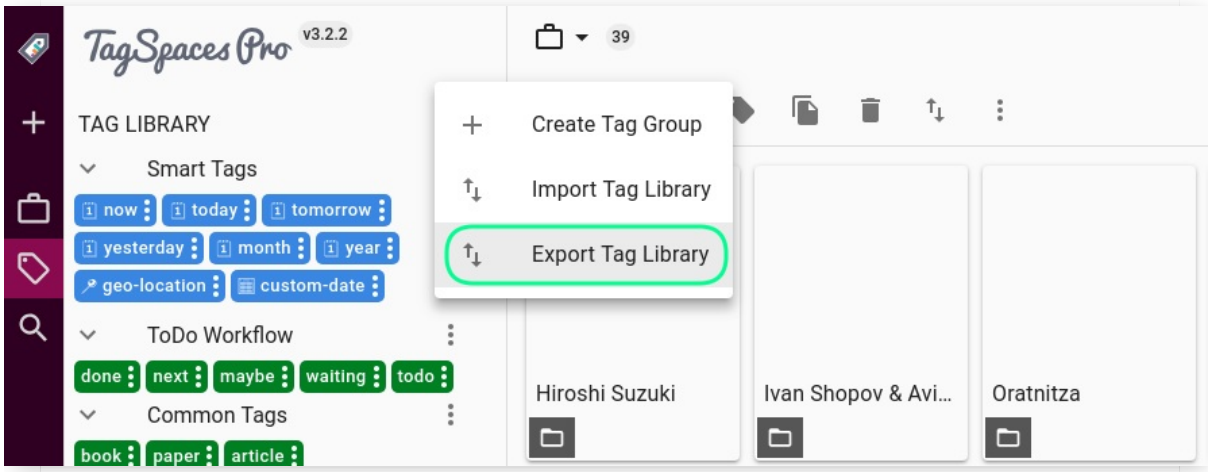
But before we start, let's define some terms from the TagSpaces jargon. **Tag Library** is a collection of tag groups, which can be opened by clicking on the button located in the bottom left cornet of the application. A **Tag Group** is a collection of tags (3). For clarification you can see the screenshot bellow.



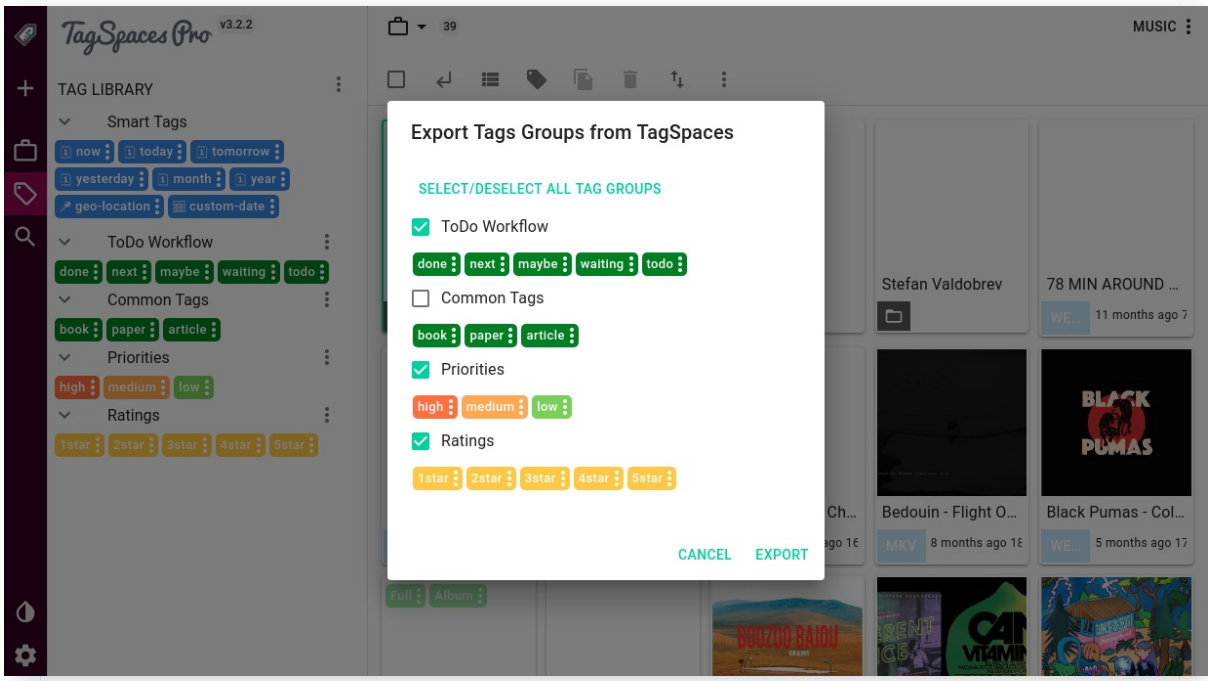
So let's assume you want share the tag groups with the names "TagGroups for Sharing 1" and "TagGroups for Sharing 2" with your colleague which is also using TagSpaces in order to have a common base for tagging on collaborative projects. As a preparation steps you can define these tag groups and put the needed tags in there, with the desired names and colors. Of course you can change them later, but at this point is much easier as you can use the TagSpaces user interface.

Exporting the existing tag groups

After adding all the needed tags and specifying their colors you are now ready to export them. Just click the three dot menu on top right area of the tag library and select "Export Tag Library".



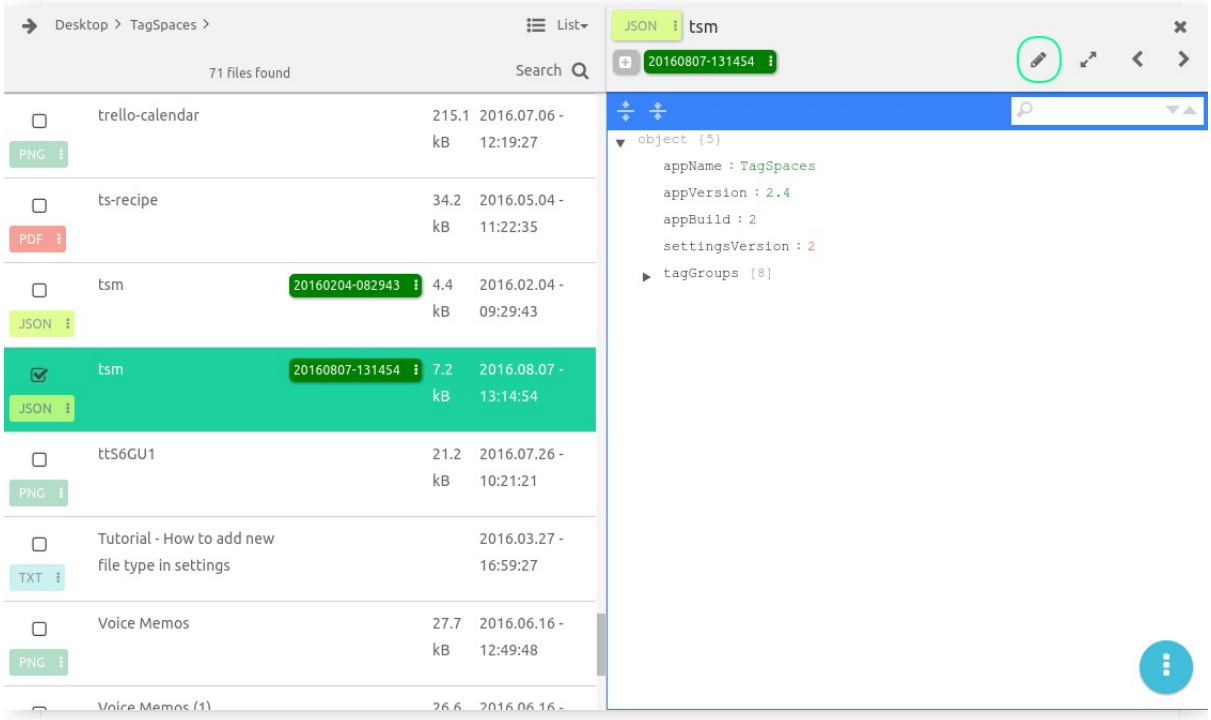
In the following screen you can select which tag group you want to export.



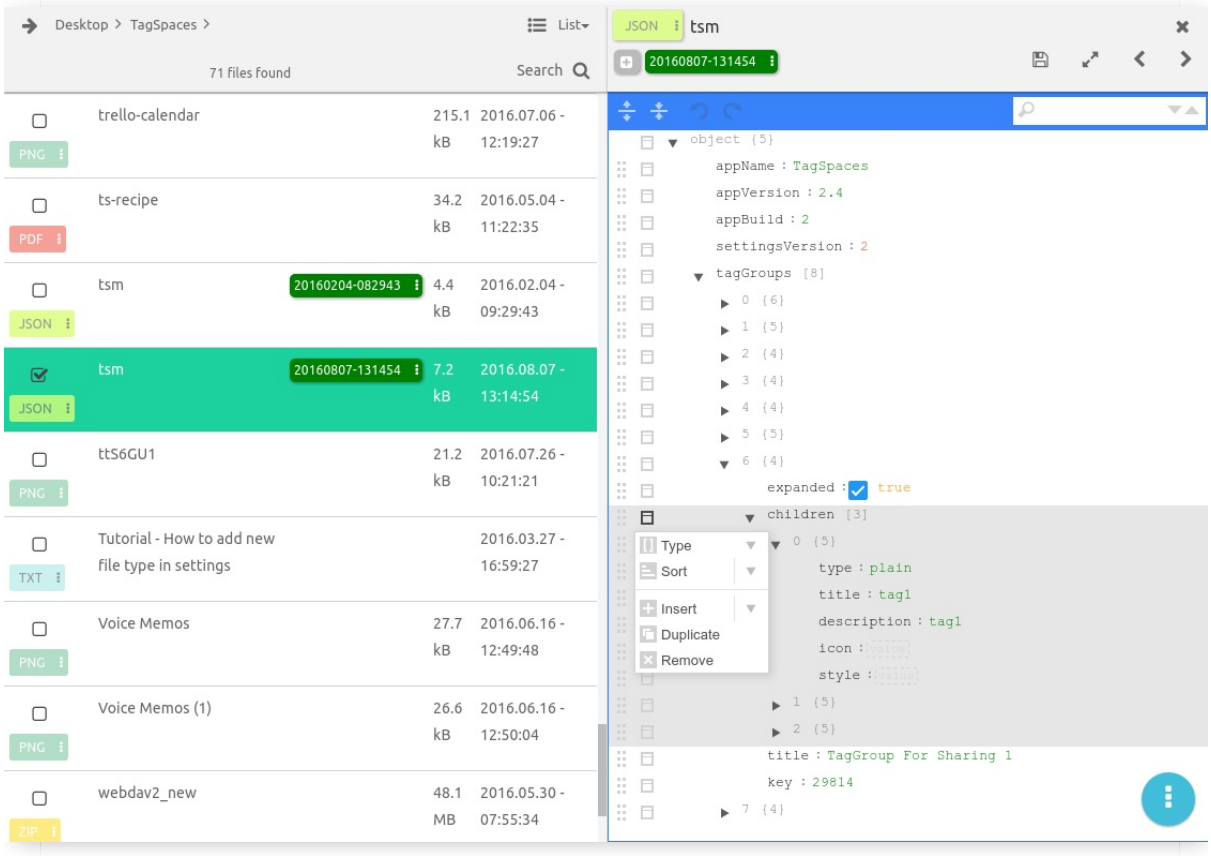
After clicking on the export button, a file saving dialog will appear, asking you to select a location where to save the file with the exported tag groups. This is files a called "tsm[20160807~131454].json", where the date/time stamp in the tags reflects the date and the time of the export. More on the export file format can be found in the file format specification.

Editing and cleaning up the exported library (optional step)

This step can be skipped if you do not need to make changes to the exported tags, otherwise you have to open the json file in some text or json editor and do the changes manually. TagSpaces has a powerful build in json editor, with which you can edit the exported json files. We assume here that we want to share only the two previously mentioned tag groups "TagGroups for Sharing 1" and "TagGroups for Sharing 2". So we will open this file with the json editor in TagSpaces and delete the unneeded tag groups or adjust some tag names. To achieve this you have to open json file with TagSpaces and click on the edit button as shown in the following screenshot.



Then you can navigate to the tag groups element and make some changes there, see the next screenshot for clarification. Following our initial intention, we have to delete all the tag groups located above the "TagGroups for Sharing 1". You can use the menu of the tag groups elements for achieving this steps. Don't forget to click in the save button in order to persist your changes.



Some other possible operation in the export file are:

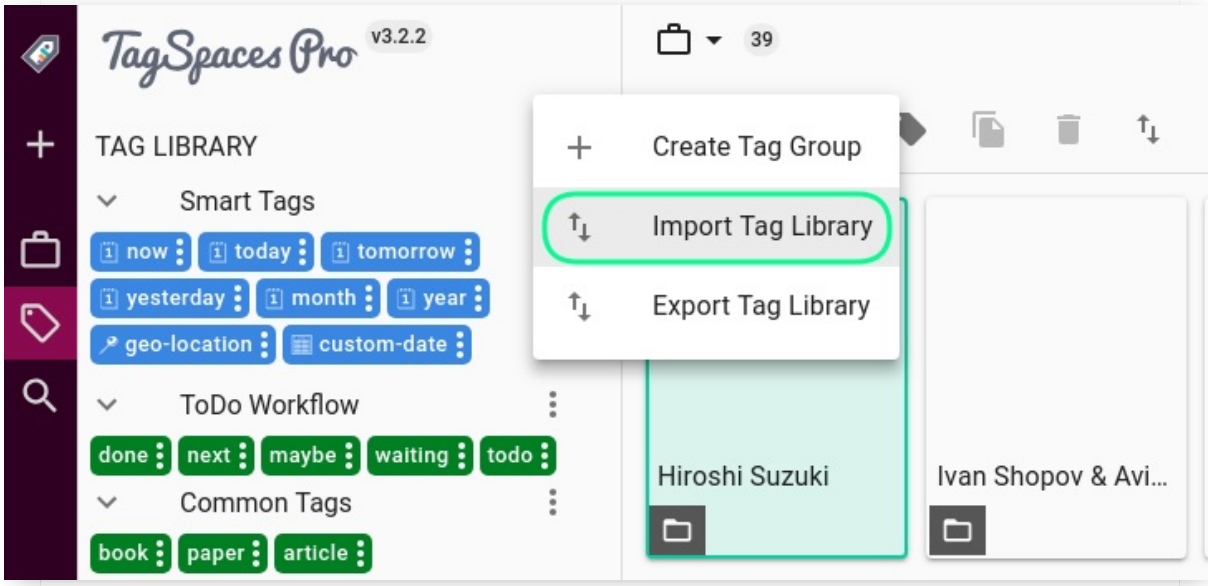
- Adding additional tags to an existing tag group
- Duplicating an existing tag group and making changes to the contained tags
- Changing the names of the tags and the tag groups
- Changing the colors of the tags

Distributing the library

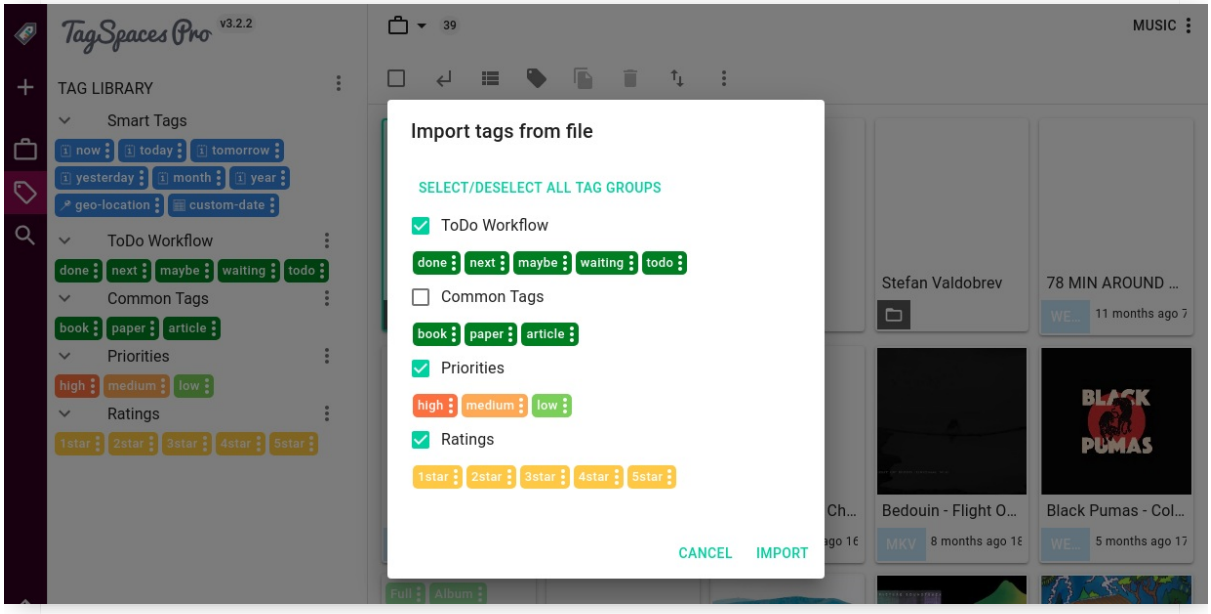
There are two main ways for distributing the exported tags, which will be discussed in the following sections.

Direct import in the tag library

In tag area of TagSpaces you will find a button, which once clicked will trigger the tags import functionality of the application.



After selecting the file and clicking the "open" button, a dialogs will appear asking you to choose the json files, which should be imported. This dialog is shown on the next screenshot.



This method has the drawback that once imported in another TagSpaces installation, these tags are disconnected from the source tag library, so any changed done in the source application, will **not** be reflected in the instance where the tags were imported.

Closing remarks

Currently you can not use any of the described methods for transferring **smart tags**, because they relay on a tighter integration in core of the application.

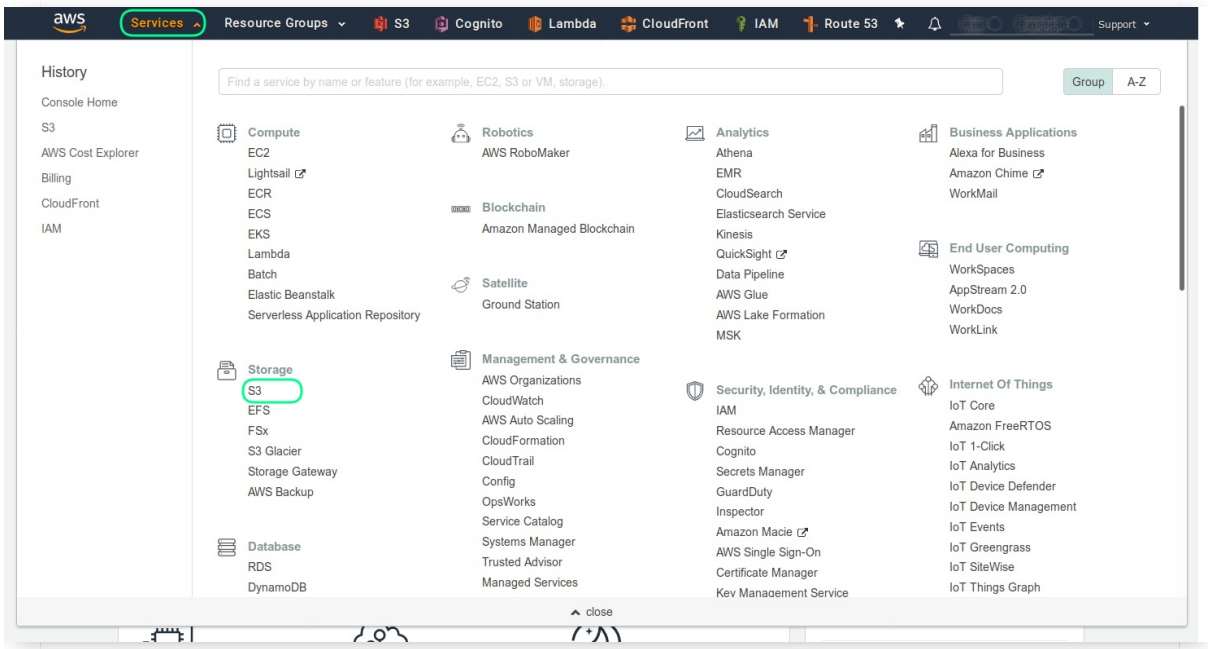
Cloud Locations in TagSpaces PRO based on AWS S3

Motivation

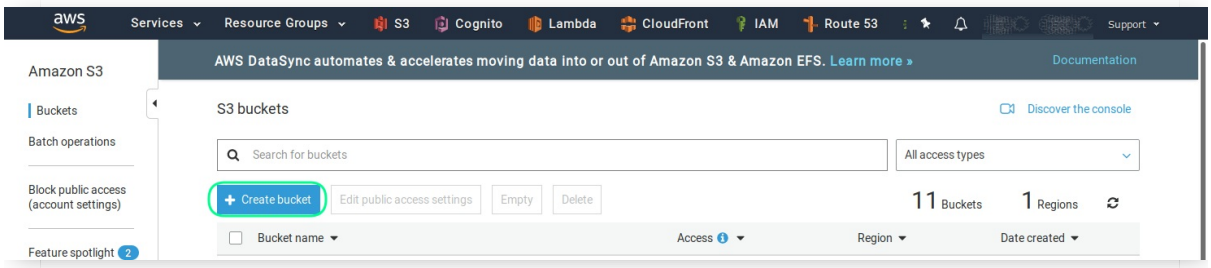
TagSpaces PRO provides the ability to connect AWS S3 compatible buckets as locations. This offers many new capabilities and use cases.

Create a bucket in AWS S3

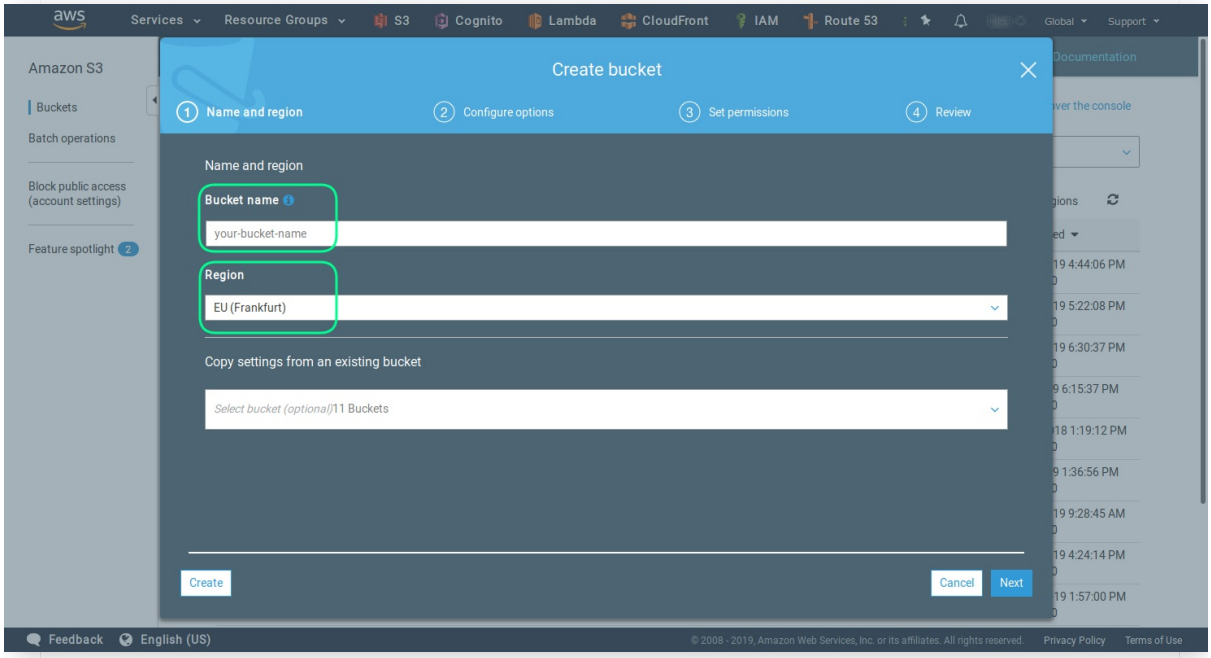
In order to create a AWS S3 bucket you have to go the Amazon Web Services website and register an account there. Once you are register and logged in go the services section and choose from there the S3, as shown in the next screenshot



And then in order to create a new bucket choose the *Create Bucket* button.

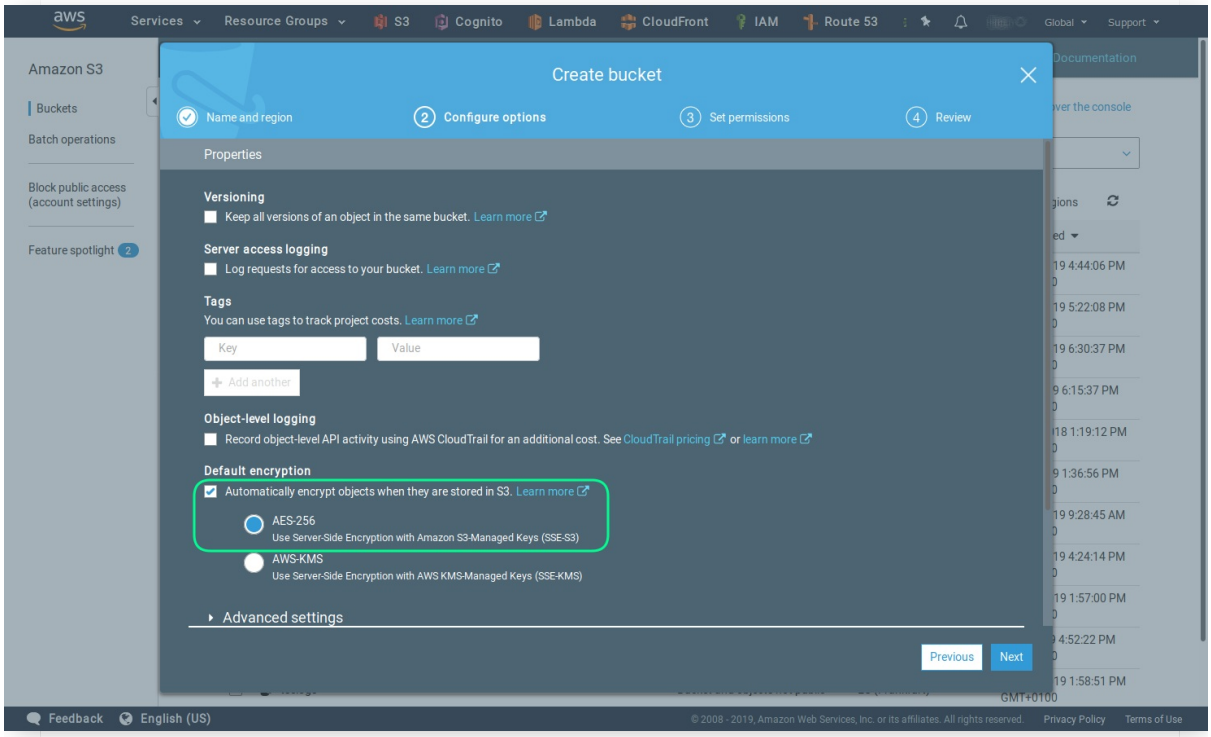


On the next screen you can choose the name of the bucket and the AWS region for the hosting. Be careful by choosing the name of the bucket, because a later rename is not possible.

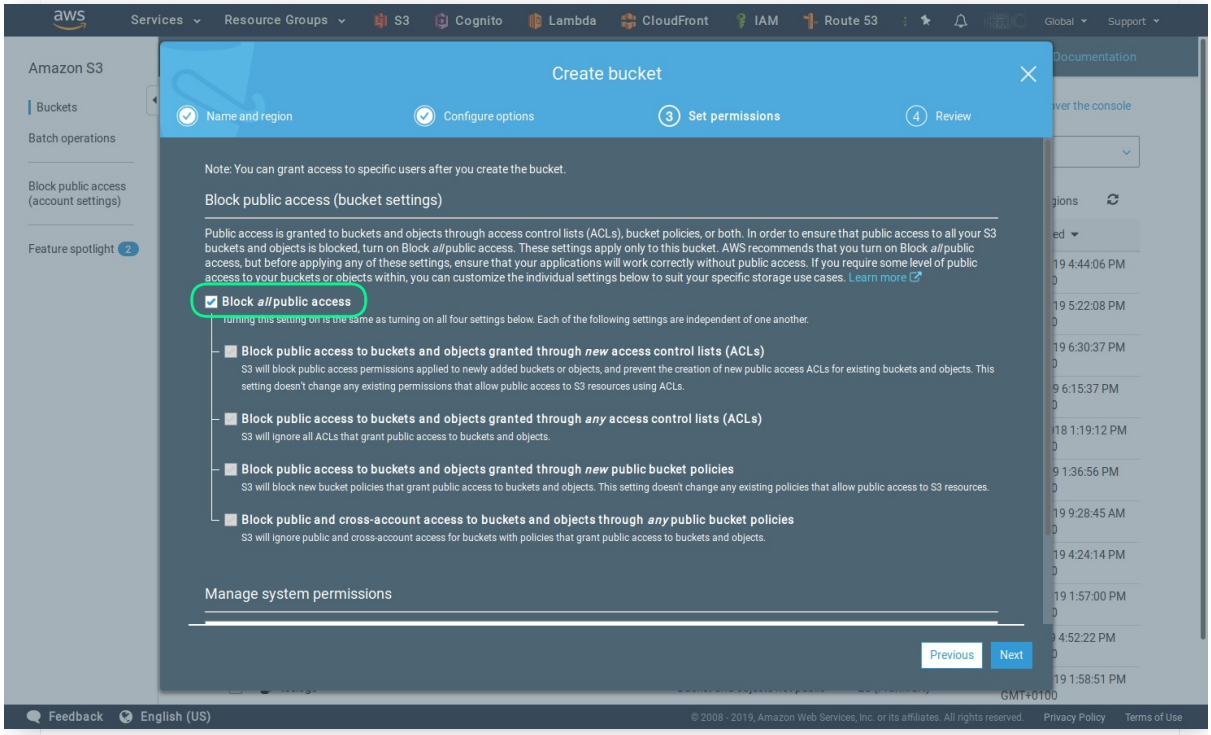


Activating the encryption is an optional, but recommended step.

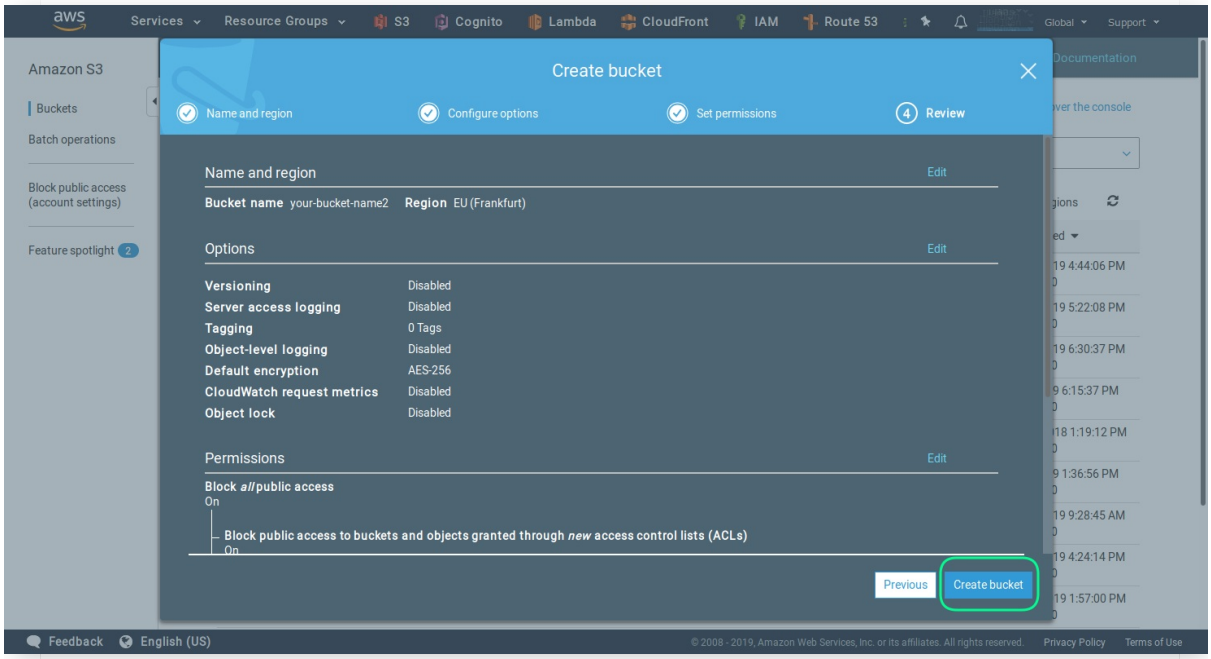
> Note that this is a server side and not an end to end encryption.



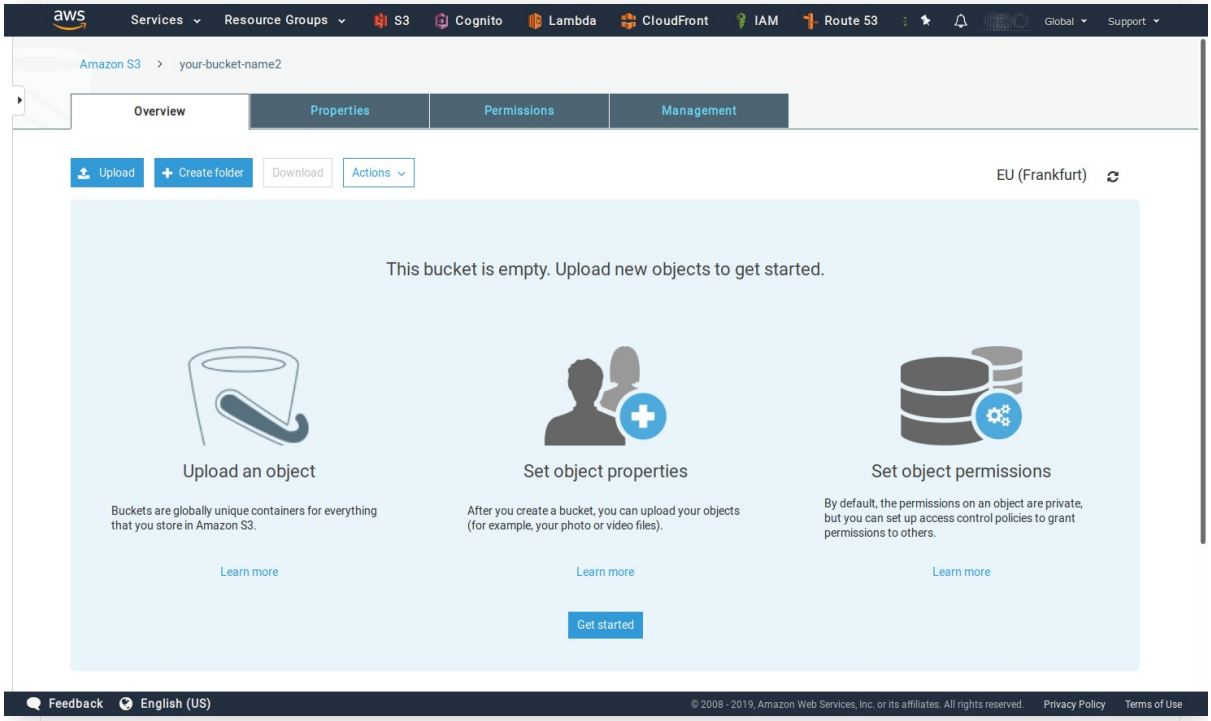
Leave the default setting for access.



Review the settings and click the *Create bucket* button.

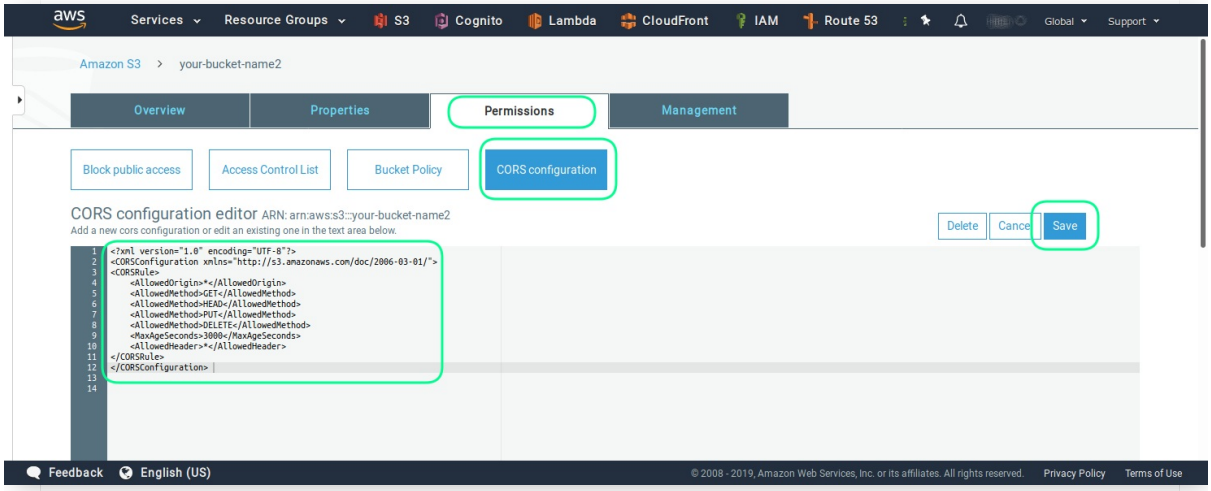


Once you have successfully created the bucket you should see the following screen.



Set the CORS settings of the bucket

This is an optional step, needed only if you want to access the bucket from TagSpaces Enterprise edition.



The XML config, can be copied from the section bellow.

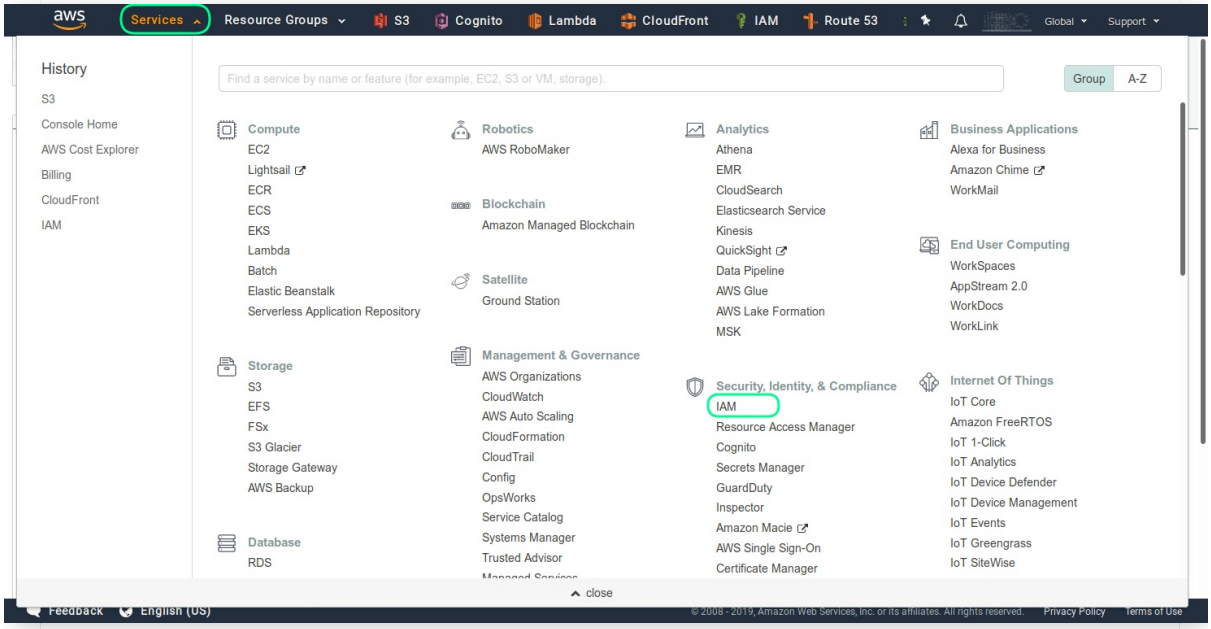
```
<?xml version="1.0" encoding="UTF-8"?>
<CORSConfiguration xmlns="http://s3.amazonaws.com/doc/2006-03-01/">
<CORSRule>
    <AllowedOrigin>*</AllowedOrigin>
    <AllowedMethod>GET</AllowedMethod>
    <AllowedMethod>HEAD</AllowedMethod>
    <AllowedMethod>PUT</AllowedMethod>
    <AllowedMethod>DELETE</AllowedMethod>
    <MaxAgeSeconds>3000</MaxAgeSeconds>
    <AllowedHeader>*</AllowedHeader>
</CORSRule>
</CORSConfiguration>
```

> You can remove the *PUT* and *DELETE* lines, if you want to disable the writing and deleting operation from TagSpaces Enterprise.
>
> It is recommended to put in the *AllowedOrigin* line, the domain from which you will access this bucket. E.g.: https://example.com

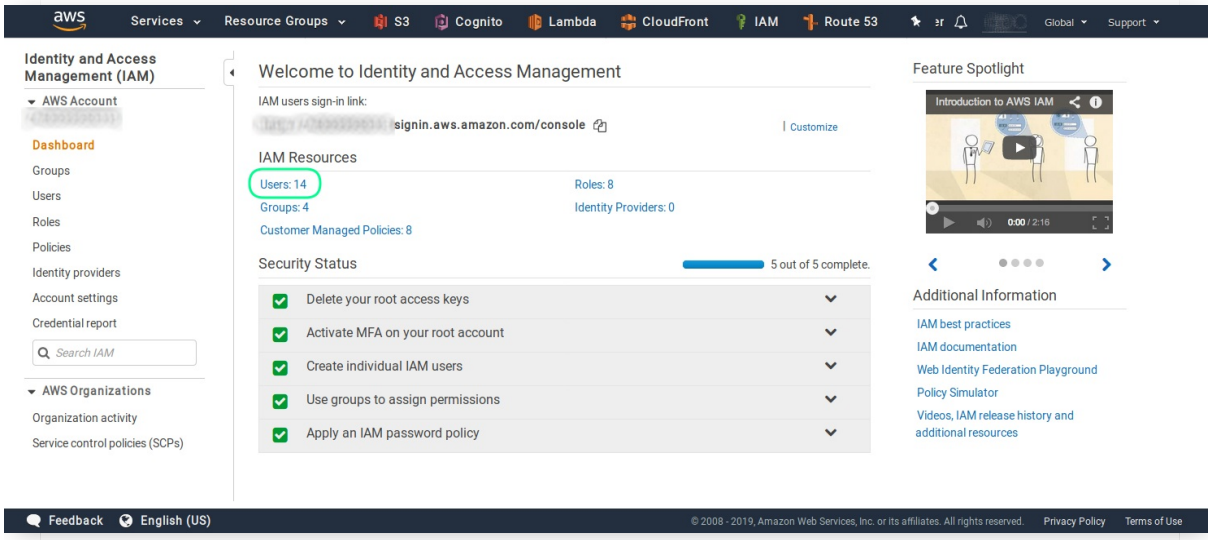
Create user for accessing the bucket.

Accessing the bucket with the credentials from your main account is not recommended. That's in this section we will guide through the process of user creation in the AWS IAM service. After successfully creating the user here, you will be able to use it for accessing the bucket from TagSpaces.

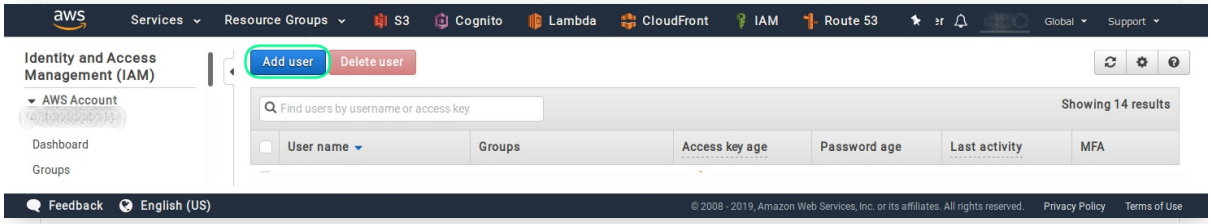
As first step the AWS IAM service should be opened.



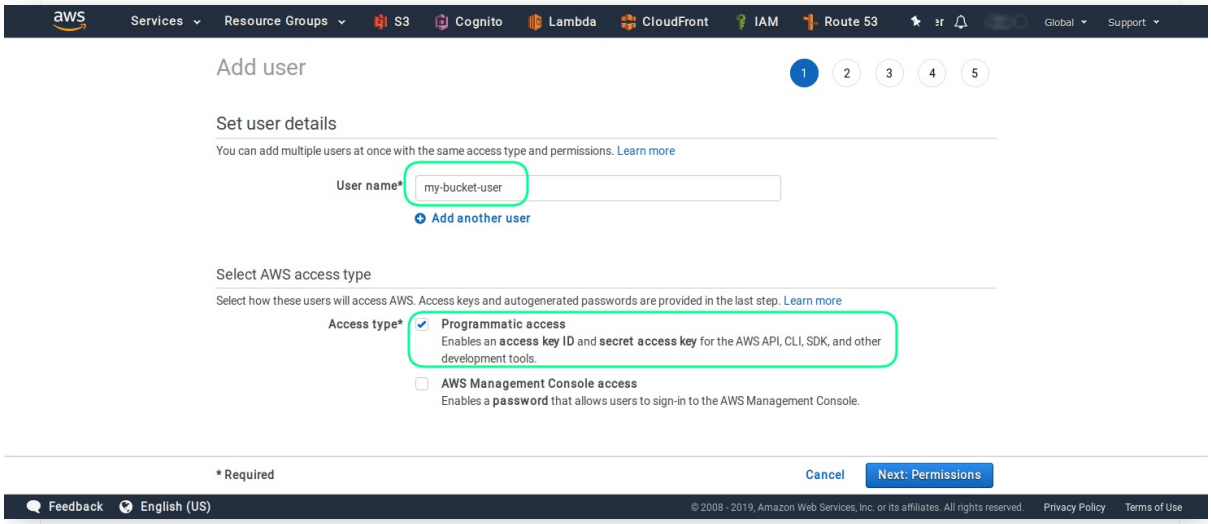
Here you can click on the **Users** section, ash shown in the screenshot.



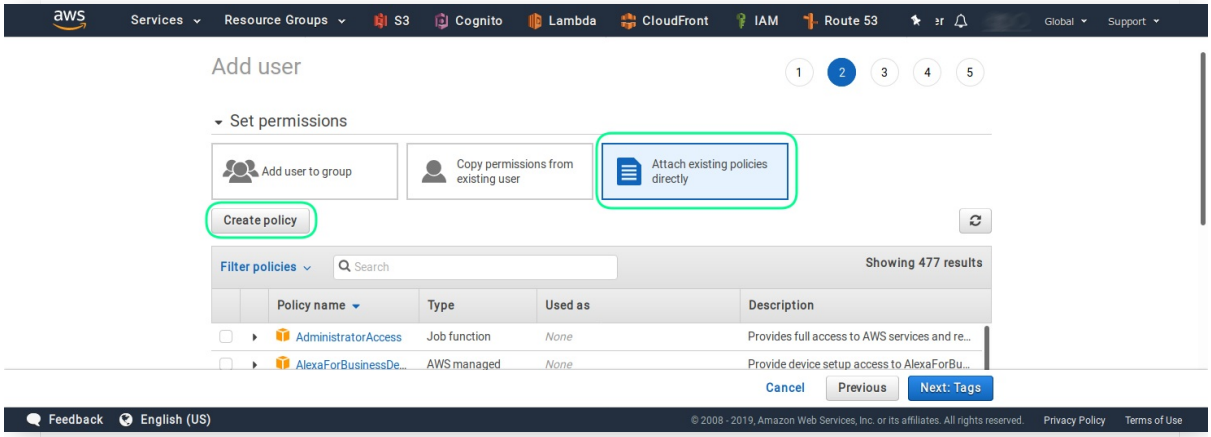
Then you have to click on the **Add user** button, in order to start the user creation process.



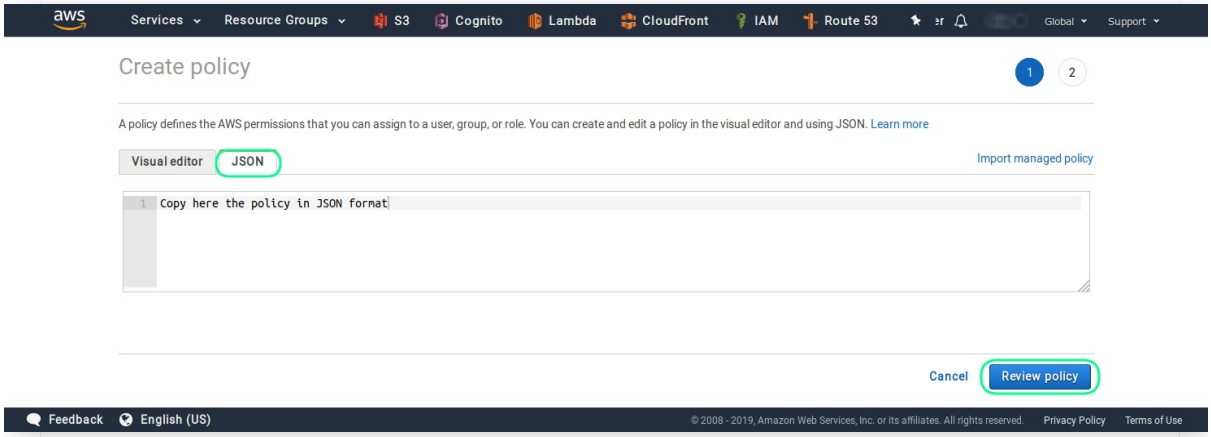
And enter the **name** and select the type of access for this user. In order to use this user for API call, you have to enable the **Programmatic access**.



In the next steps you have to set the permission for this user, by creating a custom policy. Please select the **Attach existing policies directly** and then click on the **Create policy** button.



In newly opened browser tab with the policy editor, click on the JSON section.



There you can enter and adjust your policies. The following JSON snipped, is a policy for a user who can just list and retrieve object (read-only user) from the **your-bucket-name** bucket. You should adjust the name of the bucket to suite your setup.

Policy for read-only user:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "s3:ListBucket",
                "s3:GetObject"
            ],
            "Resource": [
                "arn:aws:s3:::your-bucket-name",
                "arn:aws:s3:::your-bucket-name/*"
            ]
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "s3:GetAccountPublicAccessBlock",
            "Resource": "*"
        }
    ]
}
```

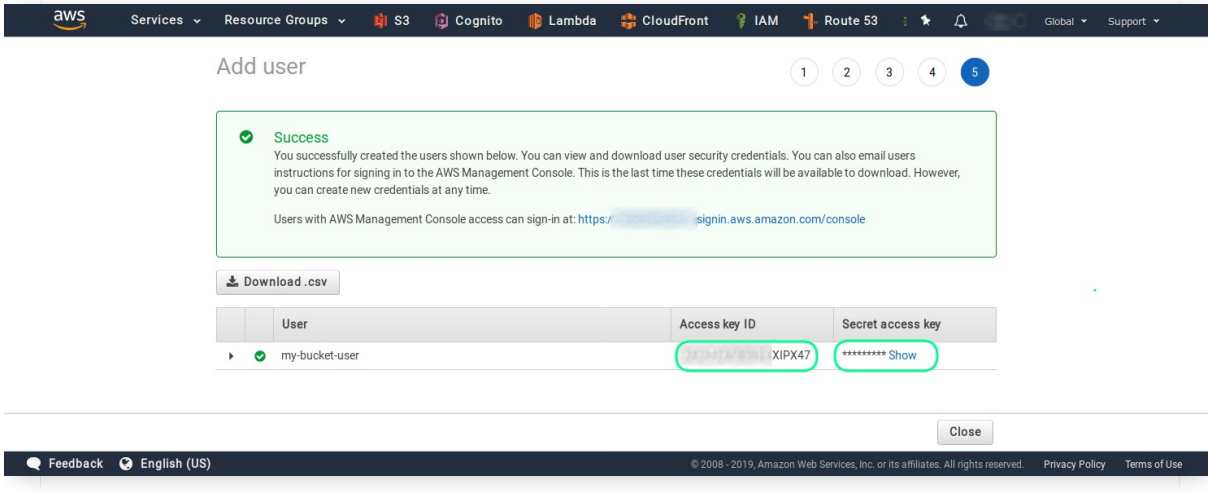
Policy for user with admin/write access:

```
{
    "Version": "2012-10-17",
    "Statement": [
        {
            "Sid": "VisualEditor0",
            "Effect": "Allow",
            "Action": [
                "s3:ReplicateObject",
                "s3:GetObjectAcl",
                "s3:GetObjectVersionAcl",
                "s3:PutObjectTagging",
                "s3:DeleteObject",
                "s3:GetBucketWebsite",
                "s3:GetBucketNotification",
                "s3:GetReplicationConfiguration",
                "s3:ListMultipartUploadParts",
                "s3:PutObject",
                "s3:GetObject",
                "s3:RestoreObject",
```

```
                "s3:ListBucket",
                "s3:GetBucketPolicy",
                "s3:GetObjectVersionTorrent",
                "s3:AbortMultipartUpload",
                "s3:GetBucketRequestPayment",
                "s3:GetObjectTagging",
                "s3:GetMetricsConfiguration",
                "s3:PutObjectAcl",
                "s3:GetBucketPublicAccessBlock",
                "s3:ListBucketMultipartUploads",
                "s3:PutObjectVersionTagging",
                "s3:GetBucketVersioning",
                "s3:GetBucketAcl",
                "s3:PutInventoryConfiguration",
                "s3:GetObjectTorrent",
                "s3:GetBucketCORS",
                "s3:GetBucketLocation",
                "s3:ReplicateDelete",
                "s3:GetObjectVersion"
            ],
            "Resource": [
                "arn:aws:s3:::your-bucket-name",
                "arn:aws:s3:::your-bucket-name/*"
                TBD
            ]
        },
        {
            "Sid": "VisualEditor1",
            "Effect": "Allow",
            "Action": "s3:GetAccountPublicAccessBlock",
            "Resource": "*"
        }
    ]
}
```

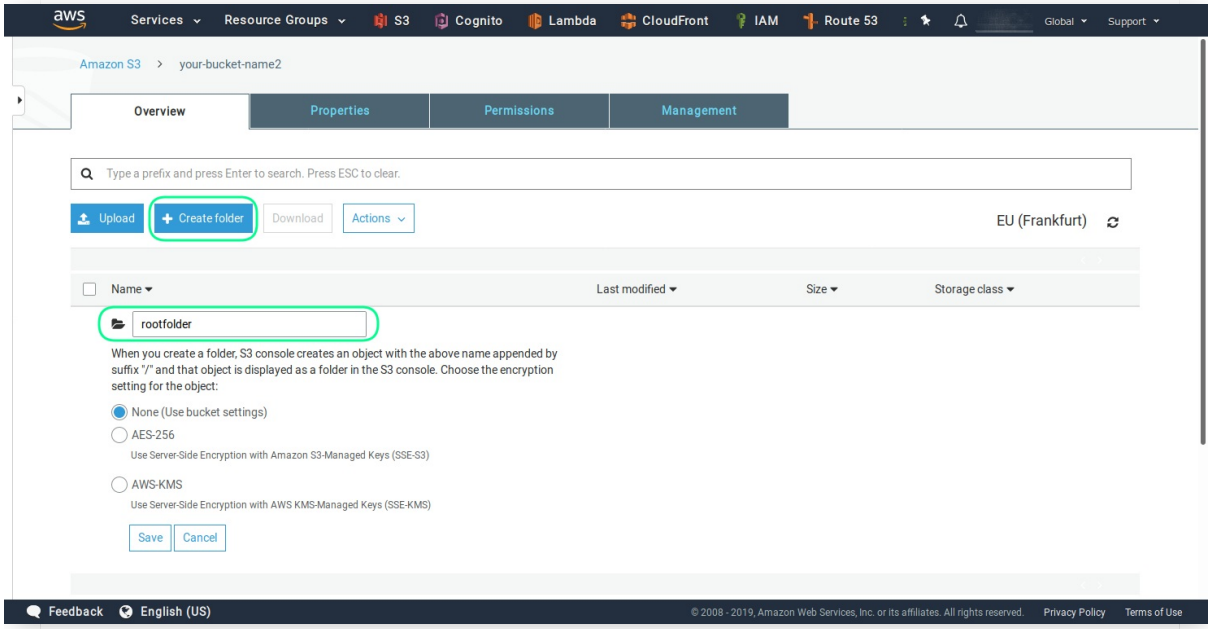
> Note: The list of the action is only a suggestion, the action can be significantly reduced, to just these which are really needed for your setup.

Once you are ready and have attached the newly created policy to the user, you can finalize the process. On the last screen you will see the **access key ID** and the **secret access key** of the just created user.

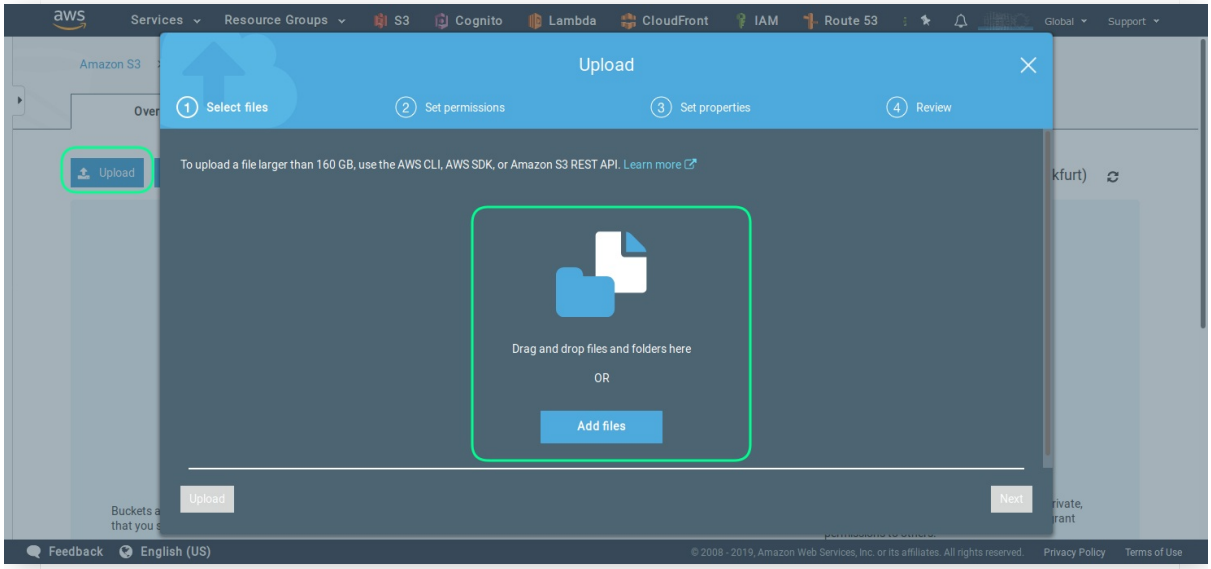


Upload files to the bucket

The easies way to upload files to your bucket is to use the build upload functionality, as seen in the next screenshot. But first you should create a folder in the bucket, which will serve as a root folder. You can name if for example *rootfolder*.



Now you can upload files using the web interface.



Alternatively you can use the AWS CLI (command line tools), with the following command.

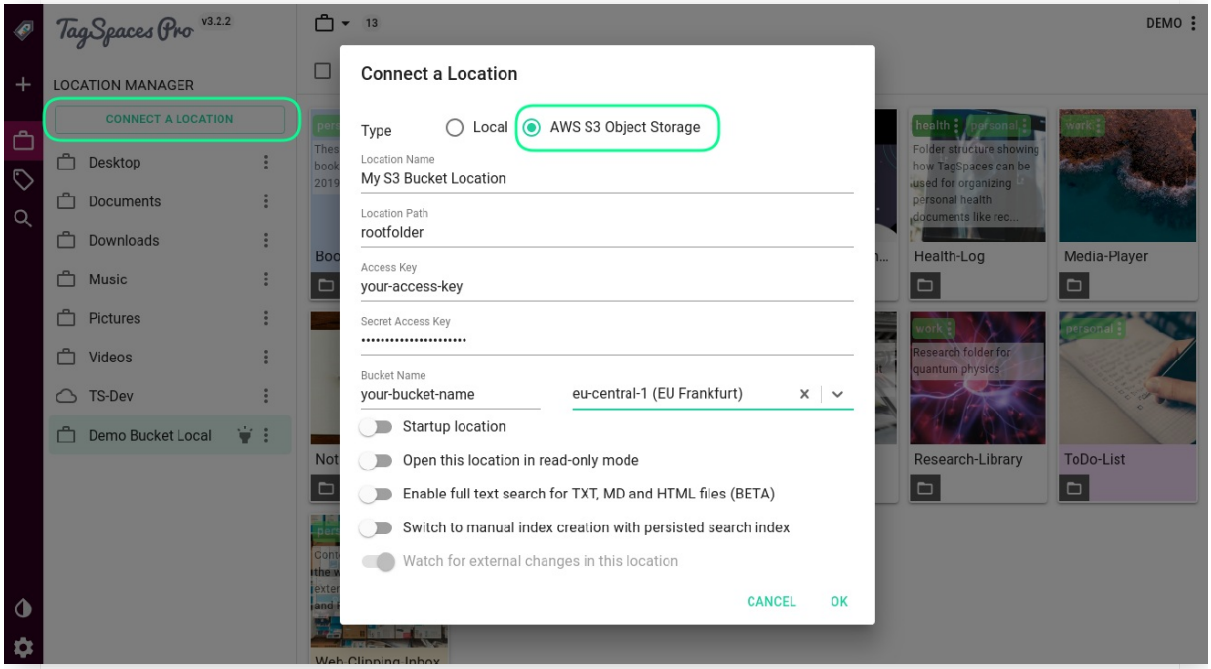
```
aws s3 sync local-bucket-folder s3://your-bucket-name/rootfolder
```

This will sync all files and folder from your local folder called *local-bucket-folder* to the sub folder with the name *rootfolder* in the bucket your bucket *your-bucket-name*

You can find out how what is *AWS CLI* and how to install it for your operating system from this link.

Create cloud location in TagSpaces PRO

Start TagSpaces and click on the **Connect a location** button from the locations section. Then you should select the *AES S3 Object Store* radio button, as shown in the following screenshot.



Here you should enter the following parameters:

- **Location Name** - this is a free text with which you will refer your location in this TagSpaces installation
- **Location Path** - is the name of the root folder from the location we have previously create
- **Access Key** - is the key of the IAM user
- **Secret Access Key** - is the secret key of the IAM user
- **Bucket Name** - self explaining ...
- **Region** - is the region of hosting for your bucket

Once you click **OK** the location will be create and its content should be listed in TagSpaces.

There are some advanced settings with can be useful for S3 locations.

- **Open this location in read-only mode** - this will switch the UI interface of TagSpaces in readonly mode. It is particularly useful for location to which the IAM user has only read-only access.
- **Switch to manual index creation with persistent index** - this option will disable the indexing of the location on its opening. Instead it will try to load the previously create index file. This is useful for large locations with many files, where the initial indexing could take a lot of time.

Setup WebDAV version

Introduction

Since a long time, i search for good Option, to save my Notes, Documents and PDF Manuals on my own Server. So far, i used Evernote but as with many other such Services, it comes the day where you think about Security and Privacy. On the other side we get spoiled with the Pleasant for the Users and no one wants live without them. In this respect, the self-hosted Service should have similar features in one form or another.

I have seen various Services for Notes self-hosting, but i found nothing for me, what makes me really happy. With some of them the Installation was really difficult, with others some important features are missing and with the rest of them, the performance on my good old Raspberry Pi2 war horrible.

Then i have found TagSpaces. It looks quite interesting, as far as the featureset is concerned. What at first deterred me, was that there it seems to be no Server Backend. So I wanted to forget TagSpaces, until I noticed that there is probably a server part, even if in some unusual form. On GitHub there is a client that can directly access a WebDAV server, such as NextCloud.

SabreDAV

I had already tested NextCloud but i wasn't happy with it. Its horrible slow and full with unnecessary functions, so that i moved to SabreDAV, since NextCloud is based on an older Version of SabreDAV. SabreDAV runs very fast on my Raspberry Pi2, as opposed to NextCloud, as long as you use a MySQL database as the Backend. There are no performance differences when comparing WebDAV from Nginx and SabreDAV. For example, the transfer of a file with 300kb other NextCloud took 14 to 50 seconds. Using the WebDAV module from Nginx, the same file takes 3 - 5 seconds. With SabreDAV it takes also only 3 - 5 seconds. SabreDAV is set on my Server that it does not use its own authentication, it relies on here with the possibilities of the Webserver. In my case, this is Nginx. The Main Reason for this is that I also use CalDAV and CardDAV over SabreDAV and Windows 10 Client's can only work with the HTTP Basic method. So you specify the following in the corresponding PHP file:

```
$authBackend = new \Sabre\DAV\Auth\Backend\Apache();
```

This Backend is also chosen if you use a different Webserver than Apache. A little further up in the PHP file you define the path to the files, which should be shared via WebDAV. This path can be defined relatively easily for each different user:

```
$publicDir = '/path_to/webdav_folder/'.$_SERVER['PHP_AUTH_USER'].'/files';
```

As shown in the example, the global variable contains $_SERVER['PHP_AUTH_USER'] username for the currently logged in User. This allows each user to use his own files. A tip: Sharing would be possible with Symlinks in this case as well. The complete PHP File could look like the following:

```
<?php
date_default_timezone_set('Europe/Amsterdam');
$publicDir = '/path_to/webdav/'.$_SERVER['PHP_AUTH_USER'].'/files';
$tmpDir = '/tmp';

$baseUri = '/dav/';

$pdo = new PDO('mysql:dbname=sabredav;host=127.0.0.1', 'mysqlusername', 'secretpassword');
$pdo->setAttribute(PDO::ATTR_ERRMODE, PDO::ERRMODE_EXCEPTION);
```

```
require_once '../vendor/autoload.php';

$authBackend = new \Sabre\DAV\Auth\Backend\Apache();

$nodes = [
    new \Sabre\DAV\FS\Directory($publicDir),
];

$server = new \Sabre\DAV\Server($nodes);
if (isset($baseUri)) $server->setBaseUri($baseUri);

$lockBackend = new \Sabre\DAV\Locks\Backend\File($tmpDir . '/locksdb');

$server->addPlugin(new \Sabre\DAV\Auth\Plugin($authBackend));
$server->addPlugin(new \Sabre\DAV\Browser\Plugin());
$server->addPlugin(new \Sabre\DAV\Locks\Plugin($lockBackend));

$server->exec();
```

In the same PHP file you can of course add other Backend's for CalDAV and CardDAV, which I have omitted in this example, because this is all about WebDAV and TagSpaces.

Nginx

Of course Nginx also needs a small adjustment. I would highly recommend to use WebDAV Sharing only via SSL (free certificates provides for example Let's Encrypt). Additionally we have to adapt the PHP handling for SabreDAV. In my case, I use a corresponding location container for my DAV services. You do not need much, my looks like this:

```
location ~ ^/sabredav/ {
        auth_basic "Secured Area";
        auth_basic_user_file /path_to/.htpasswd;

        location ~ ^(.+\.php)(.*)$ {
            try_files $fastcgi_script_name =404;
            fastcgi_split_path_info  ^(.+\.php)(.*)$;
            fastcgi_pass unix:/var/run/php/php7.0-fpm.sock;
            fastcgi_index index.php;
            fastcgi_param SCRIPT_FILENAME $document_root$fastcgi_script_name;
            include fastcgi_params;
            fastcgi_param  PATH_INFO $fastcgi_path_info;
            fastcgi_param REMOTE_USER $remote_user;
        }
    }
```

With the both auth_basic directives I set the small on Notes which appears on the Login Box in the Browser and the path to the hidden file .htpasswd. We will generate this file later. This file contains the credentials for each individual user. SabreDAV works a lot with path information, so the corresponding lines are enormously important, otherwise SabreDAV does not work as expected. Particularly important is the penultimate line fastcgi_param REMOTE_USER $ remote_user; At least for me, this was a stumbling block. As you can see, I use PHP7 from the Jessie-Backports. If you still use PHP5, the location container has to be adapted accordingly.

User Management

Now we have to create the htpasswd file or add Credentials to it if if the file already exists. I use apache-utils from Raspbian (Debian), but there are other ways to generate the credentials. Since it is a pure text file, the editing is not too difficult. With the apache-utils you create a new file with the first user as follows:

```
htpasswd -c /path_to/.htpasswd username
```

The program then asks twice for the password to be used and then stores the record in the file. If you want to change an already existing users or add a new user, leave the parameter -c simply away. If you want to delete a user, use the -d switch.

Now its time to Reload the Webserver: systemctl reload nginx

You should now be able to log in to the WebDAV URL with a browser. If this works, you have access to your shared Files. Make sure that all files belong to the Webserver-User. On Debian / Ubuntu this should be www-data. Otherwise, you may not be able to access your files for reading or writing.

TagSpaces

Let's go to TagSpaces. The installation is extremely simple. Change your path to the root directory of your Webserver. Now simply download the archive:

```
wget  https://github.com/tagspaces/tagspaces/releases/download/v2.7.0/tagspaces-2.7.0-web.zip
```

Now its time to extract the Archive:

```
unzip tagspaces-2.7.0-web.zip
```

This creates a subdirectory "tagspaces" with all the necessary files. With the Browser, you can now access the TagSpaces directory. There will be no password query, because the server knows nothing about it yet.

Nginx again

You have to take care that not everyone can access your notes. We must now protect this directory with a password. Reopen your Nginx configuration file and add a location container for the TagSpaces directory. For me, this looks something like this:

```
location /tagspaces {
        auth_basic "Secured Area";
        auth_basic_user_file /path_to/.htpasswd;
    }
```

As you can see, I use the same auth_basic variables as with the SabreDAV location. I use the same Credential File. Since these Credentials are already entered, its enough to save the changes to the Nginx Config and reload the Server: systemctl reload nginx

Now please close the browser Window so that the session ends locally. If you reopen the Browser with the URL of TagSpaces, the browser asks for the WebDAV Credentials. Enter the Credentials for WebDAV. Now you can access TagSpaces and add a new Location. In my case this is a subdirectory of the WebDAV Share. In my case sabredav/files/Notes. If you enter this Location, TagSpaces will nit ask again for Credentials since you have already established a corresponding session. From now on, you can manage your notes online via TagSpaces.

If you want, there are TagSpaces Clients for the Desktop or your Mobile Devices. Thanks to WebDAV Protocol, the files can be synchronized with any program. On Android, I use FolderSync, at Windows you can have the WebDAV share directly connected as a drive (there also exist special sync clients) and with Linux, a WebDAV Share can be seamlessly integrated into your directory structure.

Specification of the meta file formats

In comparison to many other tools, TagSpaces uses external text files for saving the meta information for folders and files, instead of a database. Here you will find the specification of the formats used by these files and also some other useful information concerning these files.

File meta description format

This file should be located in a folder called `.ts` located in the folder, where the tagged file is placed. The meta file should have exactly the same name as the tagged file, but in addition it should have the `.json` file extension. So at the end you should have similar structure as the following:

```
~ some TagSpaces location folder
├── subfolder_1
│   ├── .ts
│   │   ├── file1.jpg.json
│   │   └── file2.pdf.json
│   ├── file1.jpg
│   └── file2.pdf
├── .ts
│   ├── file3.png.json
│   └── file4.docx.json
├── file3.png
└── file4.docx
```

The meta information is saved in JSON format, which has the following format:

```
{
  "tags": [  // A set containing the tags
    {
      "title": "tag1", // The name of the tag
      "type": "sidecar", // The type of the tag
      "style": "color: #ffffff !important; background-color: #FFCC24 !important;" // CSS used for styling of th
e tag
    },
    {
      "title": "tag2",
      "type": "sidecar",
      "style": "color: #ffffff !important; background-color: #FFCC24 !important;"
    }
  ],
  "appVersionCreated": "2.4.1", // optional element, containing the version of tha app, created this file
  "appName": "TagSpaces", // optional element, containing the name of the app, created this files
  "appVersionUpdated": "2.4.1", // optional element, containing the version of the app, which last changed the
file
  "lastUpdated": "2016-06-24T12:22:38.560Z" // optional element
}
```

Folder meta description format

In the PRO version of the application you can add tags and description to every folder managed in TagSpaces. This meta information is persisted in a file called **tsm.json** located in `.ts` folder of the tagged folder. The following is an example folder structure of a tagged folder with one tagged subfolder.

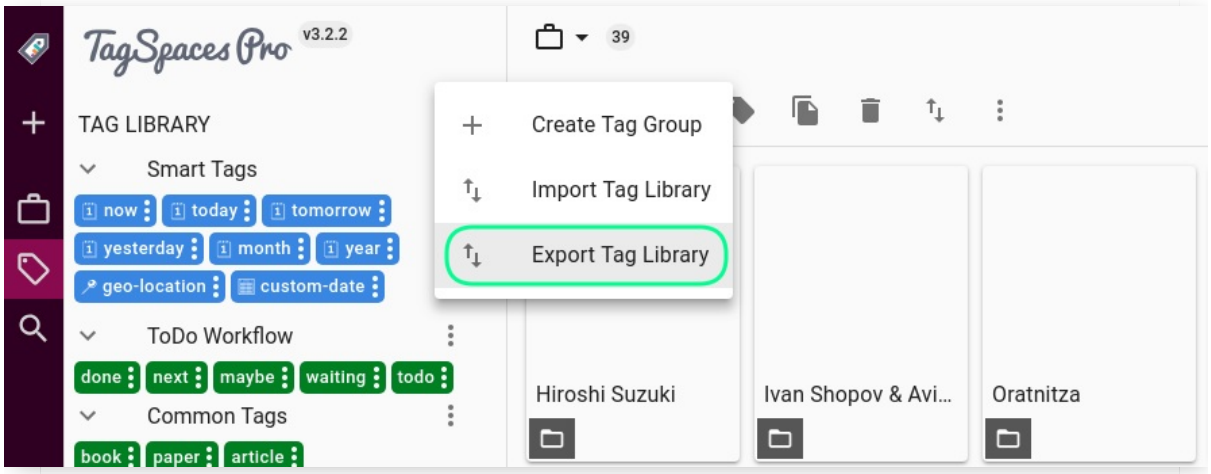
```
~ some TagSpaces location folder
├── subfolder_2
│   ├── .ts
│   │   ├── tsm.json // a file containing the meta info for subfolder_2
│   │   └── file2.pdf.json
│   └── file2.pdf
├── .ts
│   ├── tsm.json
│   └── file4.docx.json
└── file4.docx
```

The meta information is saved in JSON format, which has the following format:

```
{
  "appName": "TagSpaces", // optional element, containing the name of the app, created this files
  "tags": [ // the tags with which the folder is tagged
    {
      "title": "tag1",
      "type": "plain"
    }
  ],
  "tagGroups": [ // optional list used when the folder is selected as a location in TagSpaces, it contains the
tag groups specific to the location, which are imported in the tagging area after loading this location
    {
      "title": "TagGroup Name",
      "key": "29814",
      "expanded": true, // if false the tag group will be shown closed, hiding all the contained tags
      "children": [ // the list containing the tags of the tag group
        {
          "type": "plain", // the type of the tag
          "title": "tag1", // the name of the tag
          "description": "tag1", // not supported element
          "icon": "", // the name of the icon class from font-awesome
          "style": "" // css styling information for the tag
        },
      ]
    },
  ]
  "appVersionUpdated": "2.4.1", // optional element, containing the version of tha app, created this file
  "lastUpdated": "2016-04-05T17:12:02.237Z", // the late date and time, when the file was updated
  "description:": "Some folder description '\n' can be used for adding a new line"
}
```

Exported tags description format

All tag groups and tags can be exported from the settings of the application, as shown in the next screenshot.



The meta information is persisted in JSON format, which has the following format:

```
{
  "appName": "TagSpaces",
  "appVersion": "2.4",
  "appBuild": "2",
  "settingsVersion": 2,
  "tagGroups": [ // A list of all tag groups exported from the application
    {
      "title": "Common Tags", // the name of the tag group
      "isFolder": "true",
      "key": "OTB", // the key of the tag group
      "expanded": "true", // if false the tag group will be shown closed, hiding all the contained tags
      "children": [ the list containing the tags of the tag group
        {
          "type": "plain", // the type of the tag
          "title": "book", // the name of the tag
          "description": "tag1", // not supported element
          "icon": "", // the name of the icon class from font-awesome
          "style": "" // css styling information for the tag
        },
        {
          "title": "paper",
          "type": "plain",
          "description": "paper",
          "icon": "",
          "style": ""
        }
      ],
    },
    {
      "title": "Getting Things Done",
      "isFolder": "true",
      "key": "GTD",
      "expanded": "true",
      "children": [...] // tags removed
    },
    {
      "title": "Smart Tags",
      "key": "SMR",
      "expanded": true,
      "children": [...] // tags removed
    },
    {
      "title": "Priorities",
      "key": "PRI",
      "expanded": true,
      "children": [...] // tags removed
    }
  ]
}
```

> **Note** In the JSON examples above, you will find sometimes a description after the these characters `//`. Please note that this is not part of the format and is used only for clarification purposes.

Extension development guide

This is an initial version of a guide intended to clarify the process of extension development for TagSpaces.

Prerequisites

Cloning the TagSpaces repository from Github

```
git clone https://github.com/tagspaces/tagspaces.git
```

Setting up the development environment

Using the script `checkoutextensions.sh` or `checkoutextensions.cmd` respectively for Linux and Windows.

For Windows users, please open your Command Prompt and execute the following command: checkoutextensions.cmd For Linux users, please open your Terminal and execute the following command: sh checkoutextensions.sh

Directory structure

After running the checkout script your dev environment should have the following directory structure:

```
~ tagspaces-github-location
├── data
|   ├── assets
|   |   └── ubuntu-font
|   ├── chromium
|   ├── cordova
|   |   └── fastclick
|   ├── electron - Electron framework core
|   ├── ext
|   |   ├── editorHTML -> tagspaces-github-location/extensions/editorHTML
|   |   ├── ...
|   |   ├── perspectiveGraph -> tagspaces-github-location/extensions/perspectiveGraph
|   |   ├── ...
|   |   ├── viewerAudioVideo -> tagspaces-github-location/extensions/viewerAudioVideo
|   |   └── ...
|   ├── js
|   ├── libs
|   |   ├── bootstrap
|   |   ├── ...
|   |   └── underscore
|   ├── locales
|   |   ├── de
|   |   ├── ...
|   |   └── zh_TW
```

```
│   ├── _locales
│   │   ├── de
│   │   ├── ..
│   │   └── zh_TW
│   ├── mozilla
│   ├── node_modules
│   │   └── fs-extra
│   ├── node-webkit
│       ├── locales
│       └── node_modules
│           ├── fs-extra
│           └── trash
├── docs
├── extensions
│   ├── editorHTML
│   ├── ...
│   ├── perspectiveGraph
│   ├── ...
│   ├── viewerAudioVideo
│   └── ...
└── node_modules
```

Please note that after running the script all extension folders in `data/ext` are connected by symlinks to the extensions in the `extensions`. In this folder you will find cloned the repositories of all supported TagSpaces extension. This way you can make changes in for e.g. `extensions/viewerImage`, which will be immediately testable after running the application, because of the symlink.

Extension initialization

On application loading TagSpaces is scanning the extension folder (e.g. `data/ext`) for available extensions. So basically it is searching every sub folder for a bower file. From the bower file TagSpaces is extractiong the id and the name of the extension, which are needed later. Currently on Firefox and Chrome the available extensions are fixed in settings and not resolved at runtime.

When a given extension is needed, TagSpaces is loading a file called `extension.js` from the folder of the extension. So this file is mandatory for every extension. It loads later with *requirejs* further javascript, css or other types of files if needed.

In the most extensions like [viewerImage] or [viewerMD] the `extension.js` is creating dynamically a new IFRAME elements which loads a file called `index.html`, where the image or markdown content is displayed or manipulated.

Messaging API

In order the extension to communication with TagSpaces the *Messaging API* can be used. It is currently in definition phase and can be found unter data/js/ext.api.js

Structure of the extension

The following is the structure of a typical extension.

```
.
├── bower.json - A mandatory file
├── .bowerrc - An optional file for specifying the location of the libraries (e.g. ./libs folder)
├── extension.css
├── extension.js - the app is searching on extension loading js file with this name.
├── main.js -
├── index.html
├── libs
│   ├── exif-js
```

```
|   |   ├── bower.json
|   |   ├── ...
|   |   └── exif.js
|   ├── jquery
|   |   ├── bower.json
|   |   ├── dist
|   |   |   ├── jquery.js
|   |   |   └── jquery.min.js
|   |   └── MIT-LICENSE.txt
|   └── jquery.panzoom
|       ├── bower.json
|       ├── ...
|       └── dist
|           ├── ...
|           └── jquery.panzoom.min.js
├── LICENSE.txt
├── locales - location of the translated files from Transifex
|   ├── de_DE
|   |   └── ns.viewerImage.json
|   ├── ...
|   └── en_US
|       └── ns.viewerImage.json
└── README.md
```

Recommended structure of the bower.json

TagSpaces uses Bower as a management tool for its extension. In this section you will find out how the mandatory bower.json should look like.

```
{
  "name": "The Cool Name", <- The name of the extension, can contain spaces
  "id": "viewerHTML", <- The id of the extension, should be the same as the folder where your ext. is located
  "description": "A TagSpaces extension for ...", <- Short description of your extension
  "type": "viewer", <- The type of your extension, could be: viewer, editor or perspective
  "version": "1.0.0", <- The version of the extension
  "dependencies": {
    "jquery.panzoom": "~2.0.5"
  },
  "devDependencies": {},
  "authors": [
    "Your Name Here - http://your-optional-website-or-email.com"
  ],
  "keywords": [
    "html",
    "viewer"
  ],
  "license": "MIT",
  "main": [
    "extension.js"
  ],
  "ignore": [
    "Gruntfile.js"
  ],
  "private": true
}
```

Internationalization

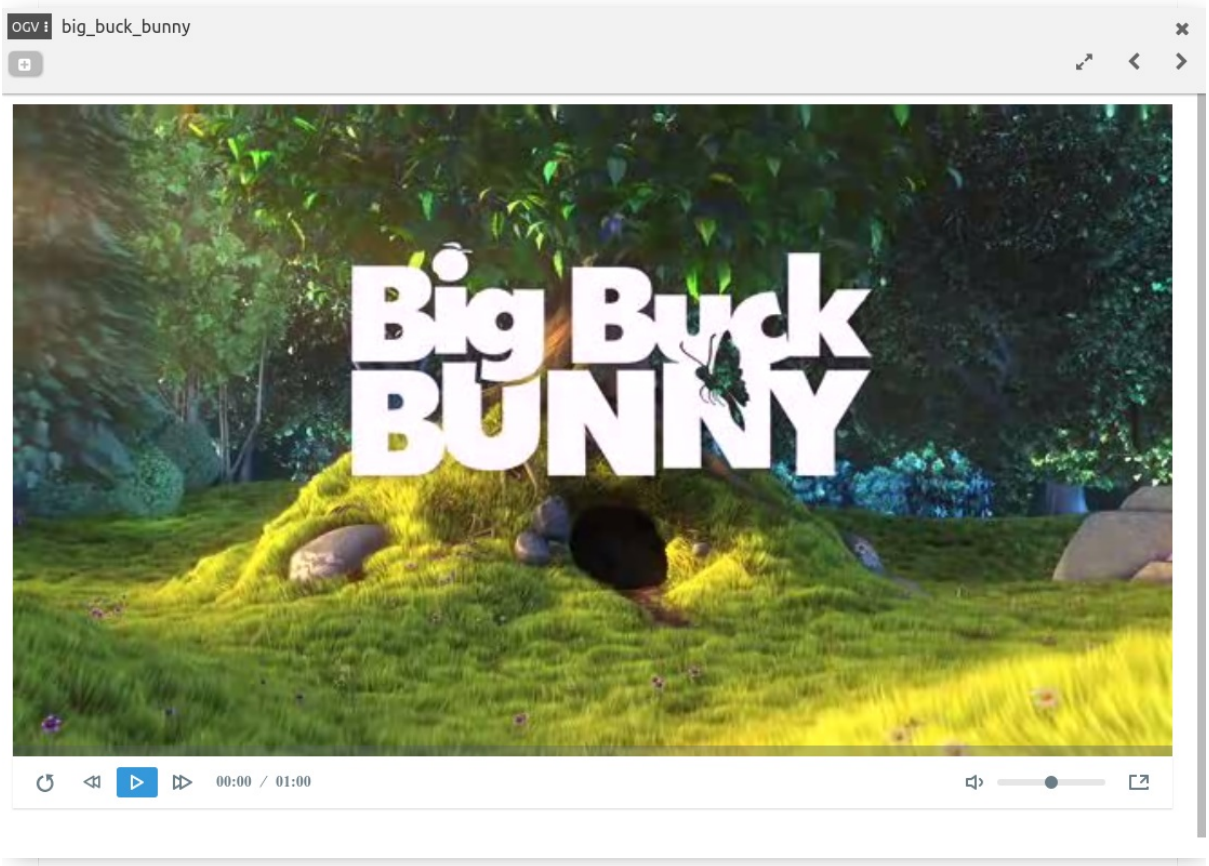
For the internationalization of the extensions we use Transifex. For some extension we have already created translation file, like for viewerImage

Audio Video Player for TagSpaces

A TagSpaces extension allowing playing of audio and video files.

Features

- Playing audio and video files
- Fullscreen mode
- This extension will trigger the opening of the next file in the current folder on finishing playback, allowing you for example to listen many audio files one after another like in a regular music player.



> The supported audio and video formats depend on the underlaying browser platform.

Used libraries

This extension thankfully relays on the following great libraries:

- plyr

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerAudioVideo

Development

If you want to extend this extensions, please follow our general extension development guide

License

MIT

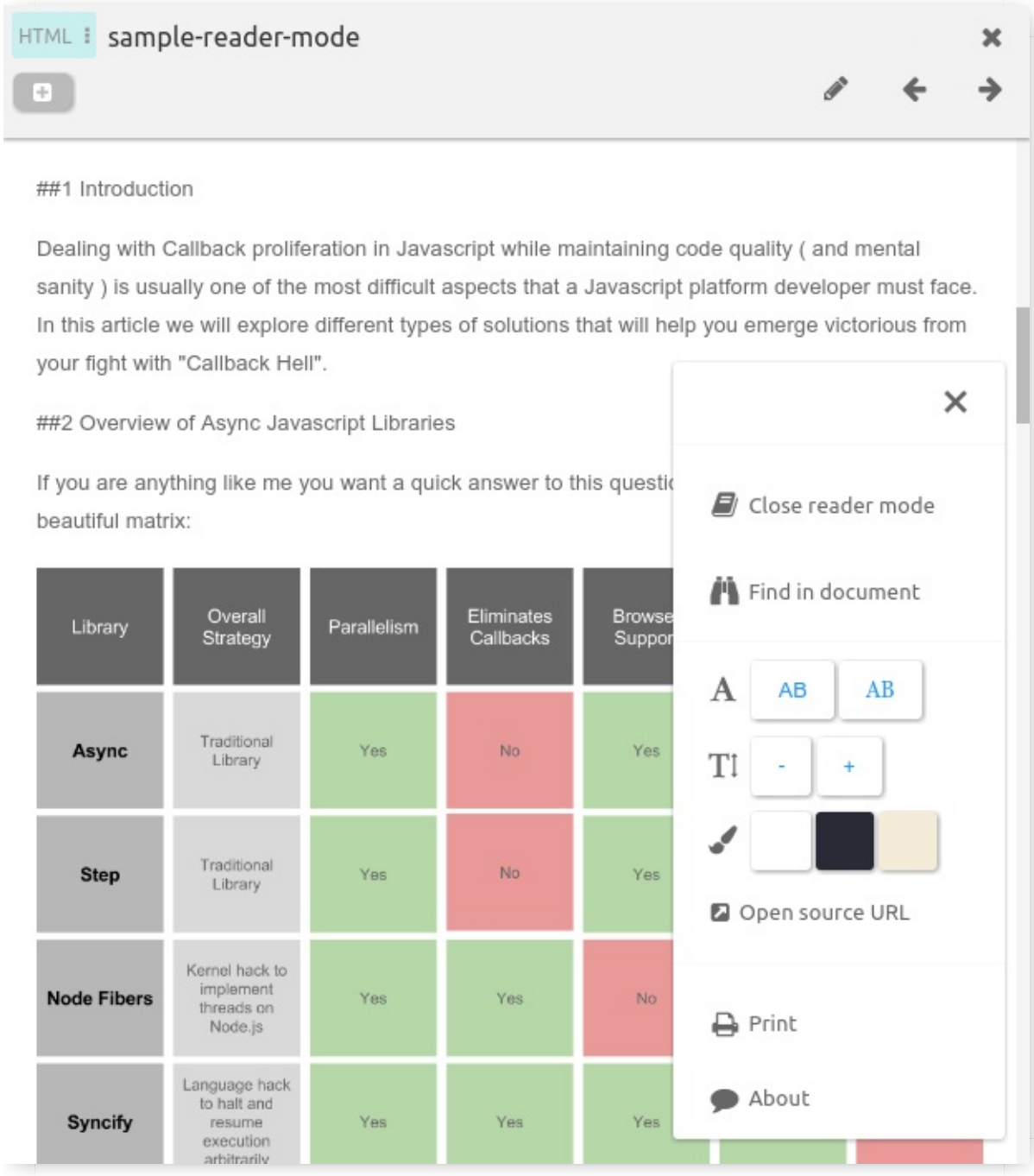
HTML Viewer for TagSpaces

A TagSpaces extension allowing opening of HTML files.

Features

Here is a list of the key feutures offered by this extension.

- Offers different themes for displaying of the documents
- Zoom in and zoom out capabilities
- Printing the document
- Search for text in the current document
- Reader mode



Meta-Data

The TagSpaces Firefox and Chrome web clippers are saving the date and time of the clipping and also the url from which the webpage is saved. If this informatation is available in the current HTML file, the menu item `Open source URL` will be displayed in extension main menu, allowing you to open the URL in a external browser.

Used libraries

This extension thankfully relays on the following great libraries:

- markdown-css
- readability

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerHTML

Development

If you want to extend this extensions, please follow our general extension development guide

License

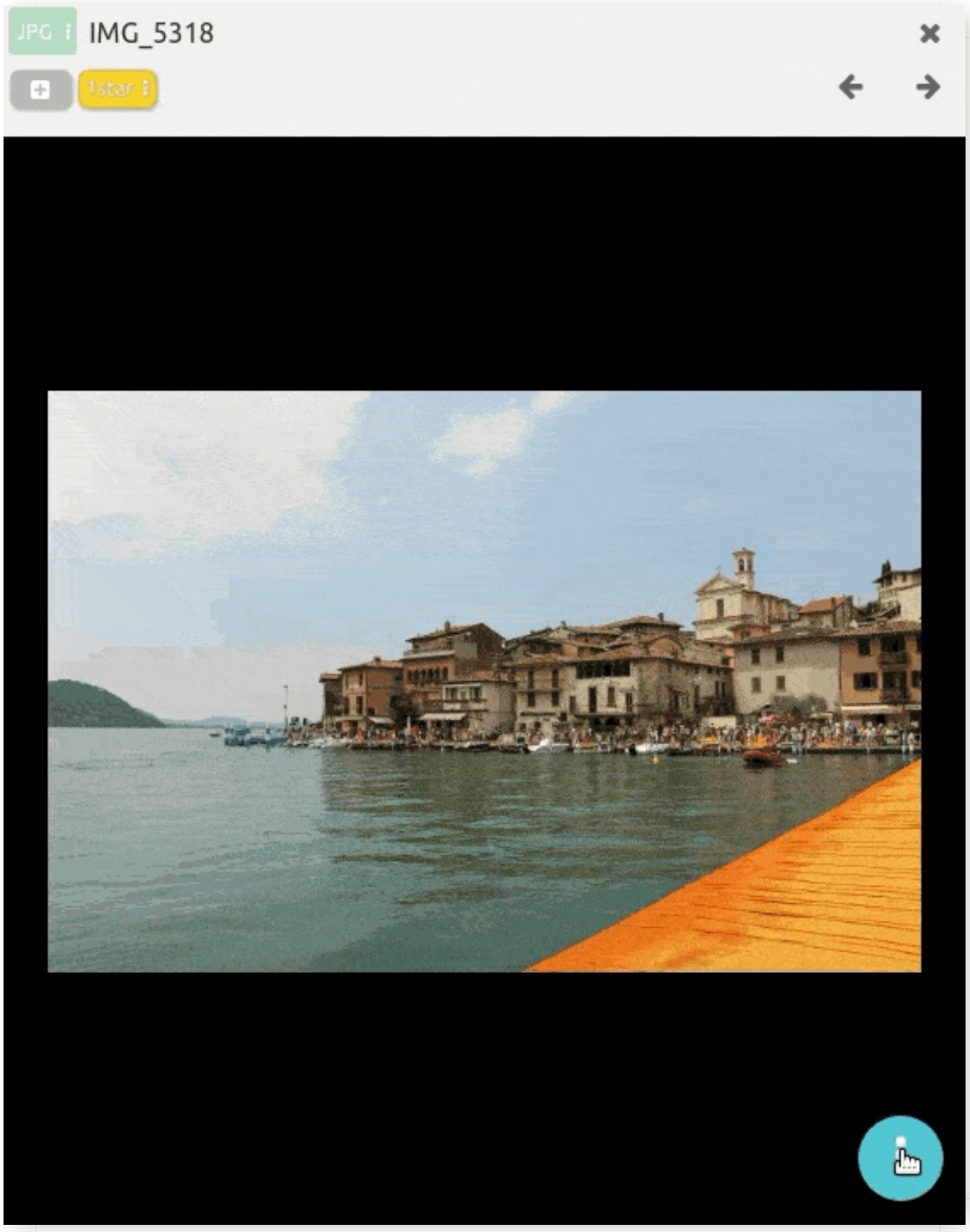
MIT

Image Viewer for TagSpaces

A TagSpaces extension allowing you to open different kinds of image formats

Features

- Opening of the following image formats: JPG, PNG, GIF, SVG, BMP, WEBP, ICO, PSD, TIFF
- Zooming, flipping and rotating of the current image
- Different background colors of the viewer for better contrast
- For JPG files it features an integrated Exif and IPTC reader with auto rotation of photos according to the Exif information
- Support image printing
- Grayscale filter



Used libraries

This extension thankfully relays on the following great libraries:

- viewer.js
- exif.js

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerImage

Development

If you want to extend this extensions, please follow our general extension development guide

License

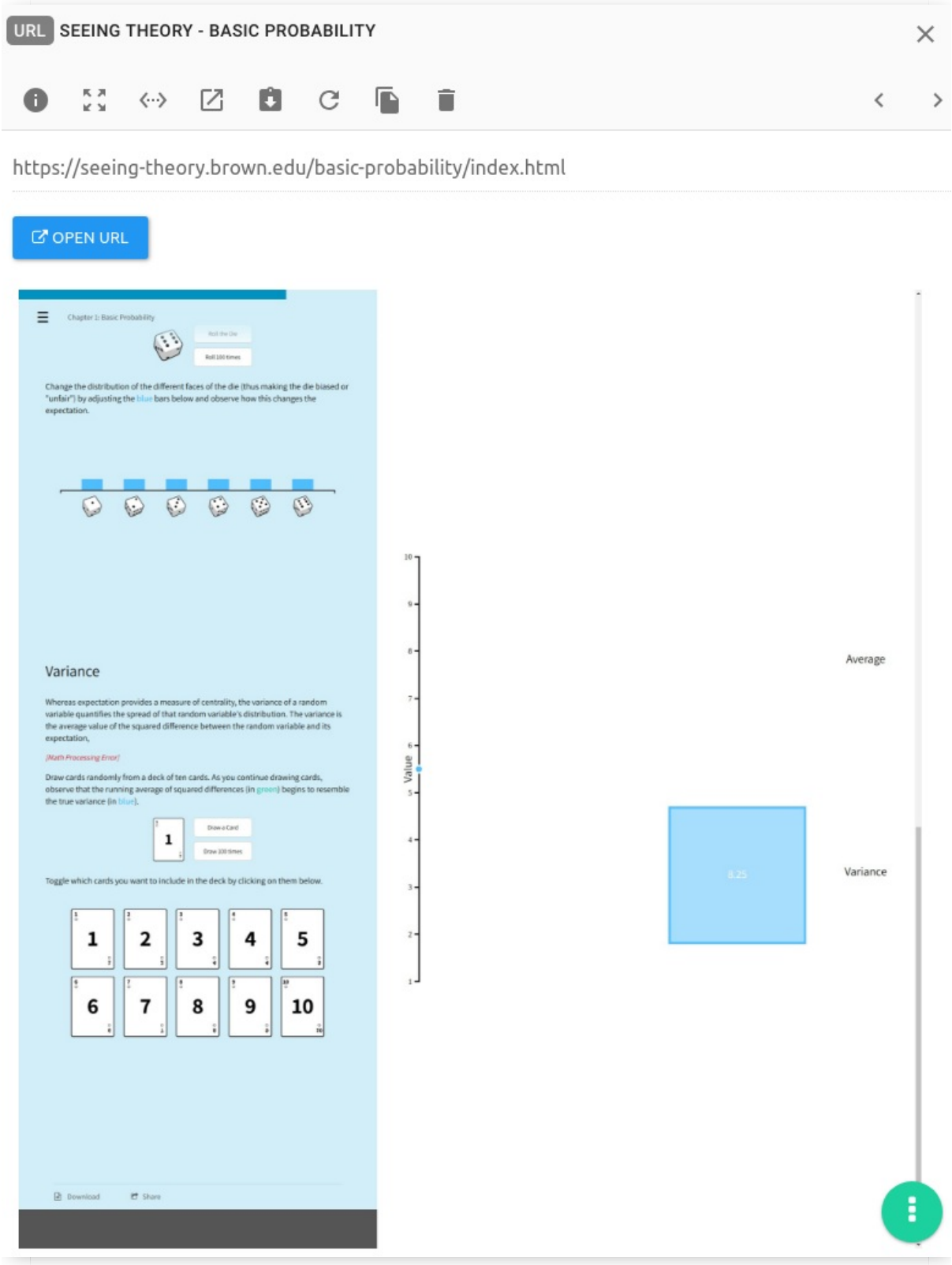
MIT

URL Viewer for TagSpaces

A TagSpaces extension allowing you to open url files created from web-browsers or bookmark manangers

Features

- Enables TagSpaces to be used as bookmark management application
- Opening the links from Ubuntu's `.desktop` files, created for example by dragging URLs from the Chrome browser
- Opening the links from Window's `.url` files, created for example by Favorite Manager of Internet Explorer
- Opening the links from Ubuntu's `.website` files
- With this extensions it possible to use TagSpaces as a bookmark manager with support of tagging.



Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerURL

Development

If you want to extend this extensions, please follow our general extension development guide

License

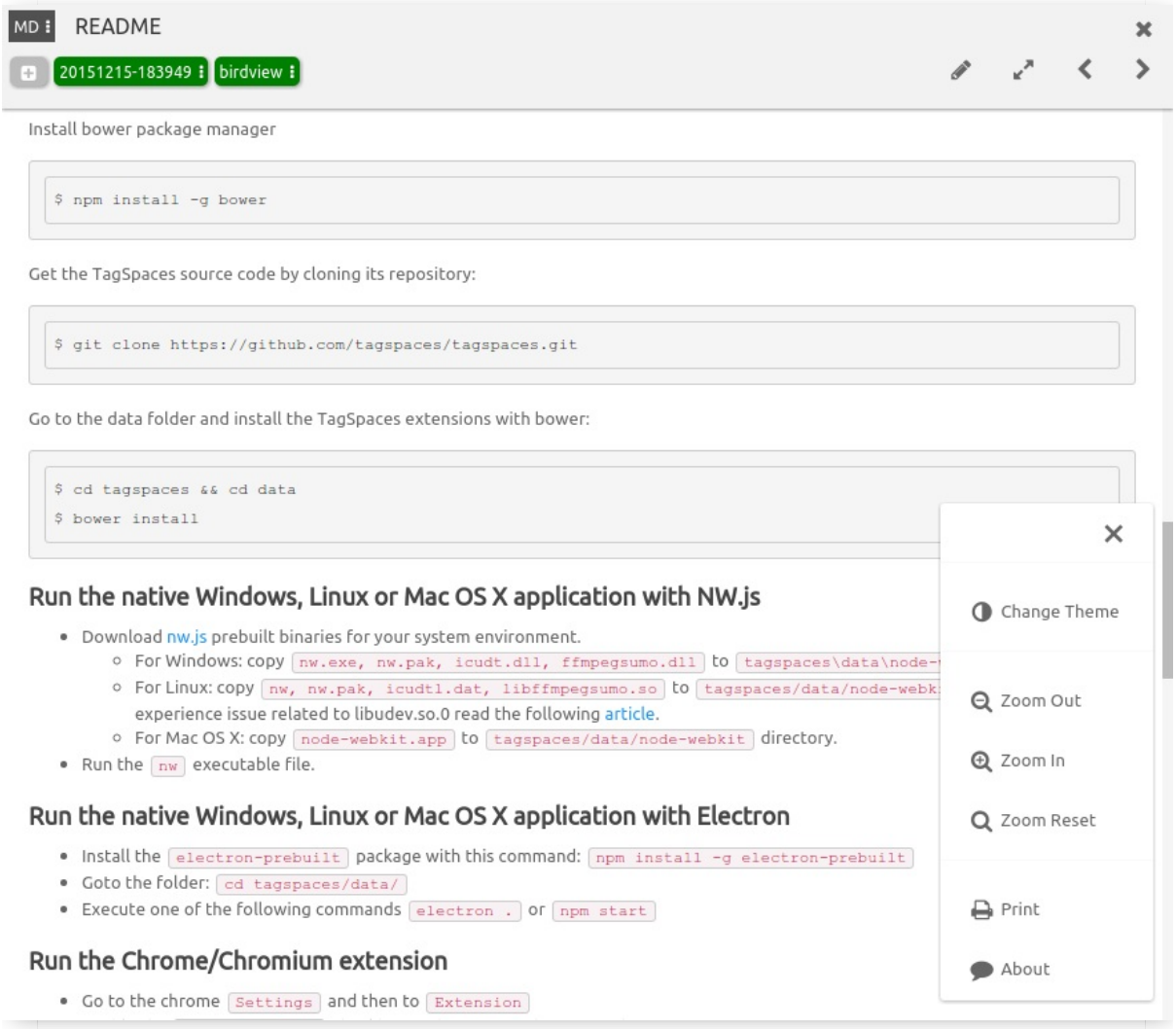
MIT

Markdown Viewer for TagSpaces

A TagSpaces extension allowing opening of markdown files

Features

- Offers different themes for displaying of the documents
- Zoom in and zoom out capabilities
- Finding text in the current document
- Document printing



Used libraries

This extension thankfully relays on the following great project(s):

- marked
- markdown-css

Installation

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerMD

Development

If you want to extend this extensions, please follow our general extension development guide

License

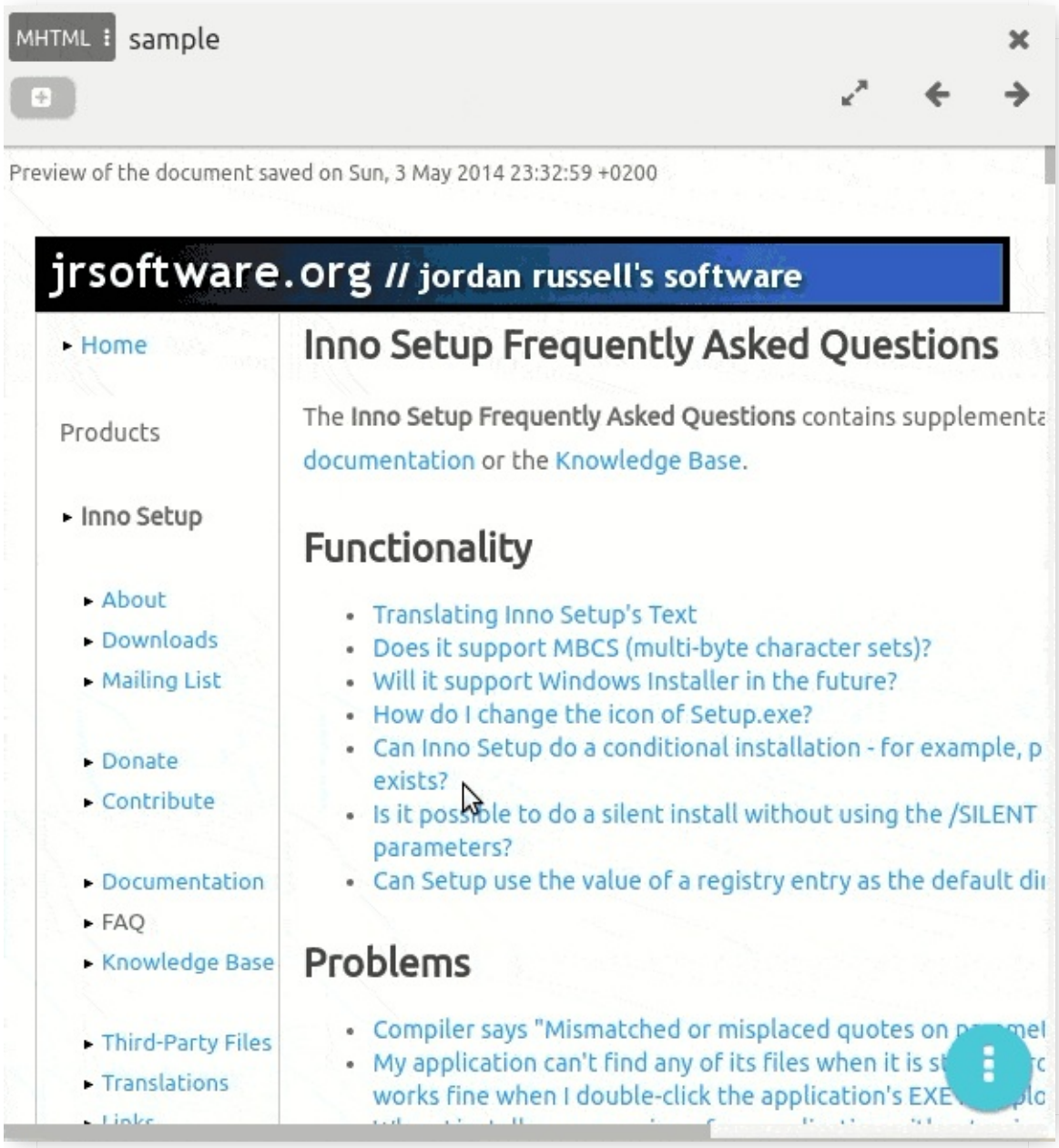
MIT

MHTML Viewer for TagSpaces

A TagSpaces extension allowing you to open MHTML and EML files.

Features

- Viewing of MHTML/MHT files - MHTML is file format for saving web pages with all the images and styling information in one single file. Saving in MHTML format is natively supported by Chrome™, Internet Explorer™ and Firefox™ (with the help of the MAFF addon) browsers.
- Viewing of EML files - EML is file format for saving emails. It is the default export email format of the Thunderbird email client. The *show original* email functionality in Gmail™ also exports the email in this format.
- Readabilty mode - for easy reading of the text content, with support for *serif* and *sans serif* font, different font size and background colors.
- Showing the creation date of the file
- Opening the source url of a MHTML file
- Finding text in the current file
- File printing



Used libraries

This extension thankfully relays on the following great project(s):

- mailparser
- readability

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerMHTML

Development

If you want to extend this extensions, please follow our general extension development guide

License

MIT

ODF Viewer for TagSpaces

A TagSpaces extension allowing opening of Open Document Format (ODF) based documents like ODT, ODP or ODS.

Features

- Viewing text documents (ODT)
- Viewing presentation documents (ODP)
- Viewing spreadsheet documents (ODS)

Used libraries

This extension thankfully relays on the following great libraries:

- webodf.js

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/editorODF

Development

If you want to extend this extensions, please follow our general extension development guide

License

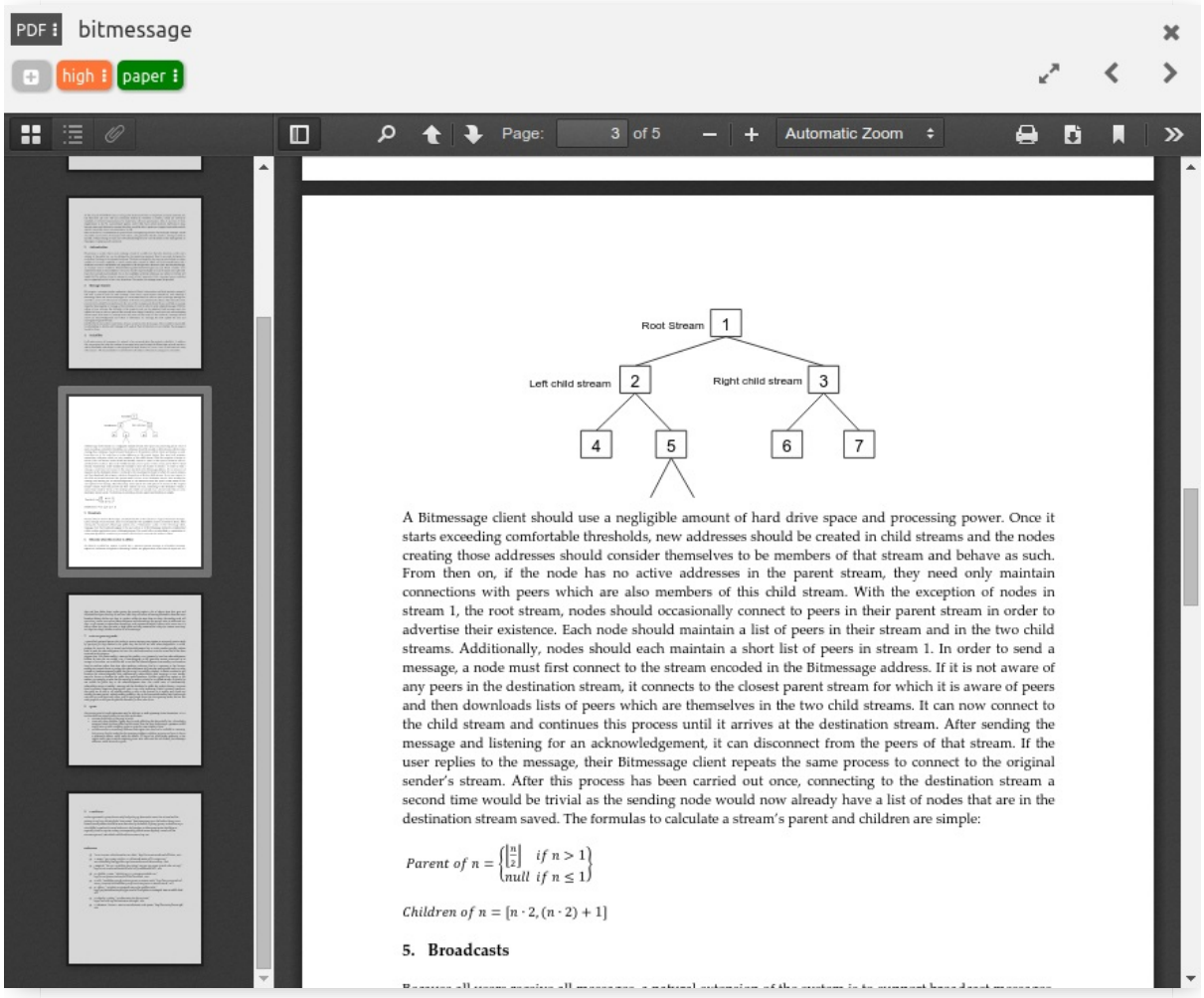
AGPL

PDF Viewer for TagSpaces

A TagSpaces extension allowing opening, searching and navigating thought PDF files.

Features

- Opening PDF documents
- Rotating and zooming of pages
- Finding text in PDF documents
- Printing PDF documents



Used libraries

This extension thankfully relays on the following great project(s):

- pdf.js

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerPDF

Development

If you want to extend this extensions, please follow our general extension development guide

License

MIT

RTF Viewer for TagSpaces

A TagSpaces extension allowing opening of RTF files.

Features

Here is a list of the key feutures offered by this extension.

Used libraries

This extension thankfully relays on the following great libraries:

- rtf.js

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerRTF

Development

If you want to extend this extensions, please follow our general extension development guide

License

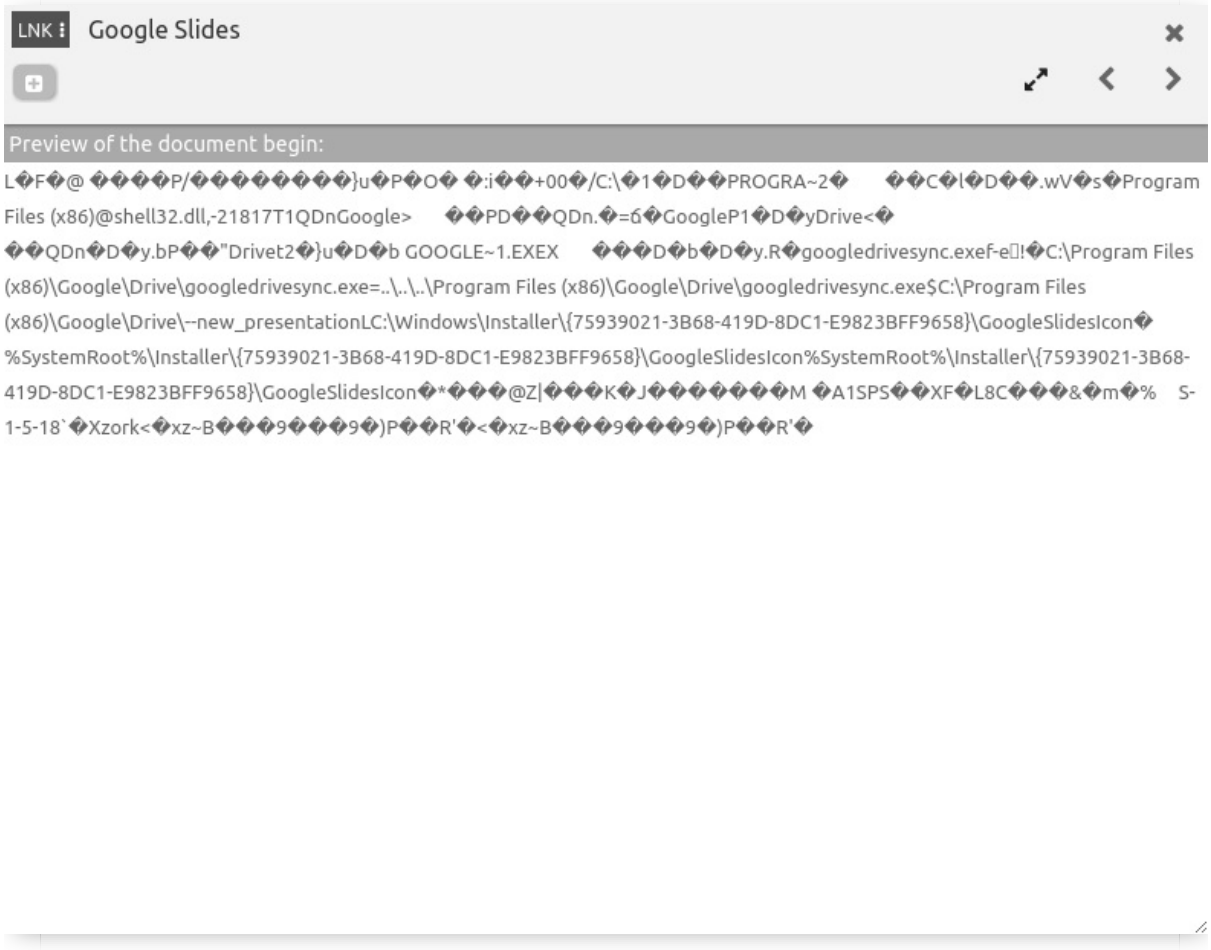
MIT

Text Viewer for TagSpaces

A TagSpaces extension allowing you to open plain text files

Features

- Displaying the content of files as plain text



Used libraries

This extension has no external dependencies

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerText

Development

If you want to extend this extensions, please follow our general extension development guide

License

MIT

General Viewer for TagSpaces

A TagSpaces extension allowing to open files in the embedded browser.

Features

- opens various file format supported in the underlbrowser

Used libraries

no dependencies

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerBrowser

Development

If you want to extend this extensions, please follow our general extension development guide

License

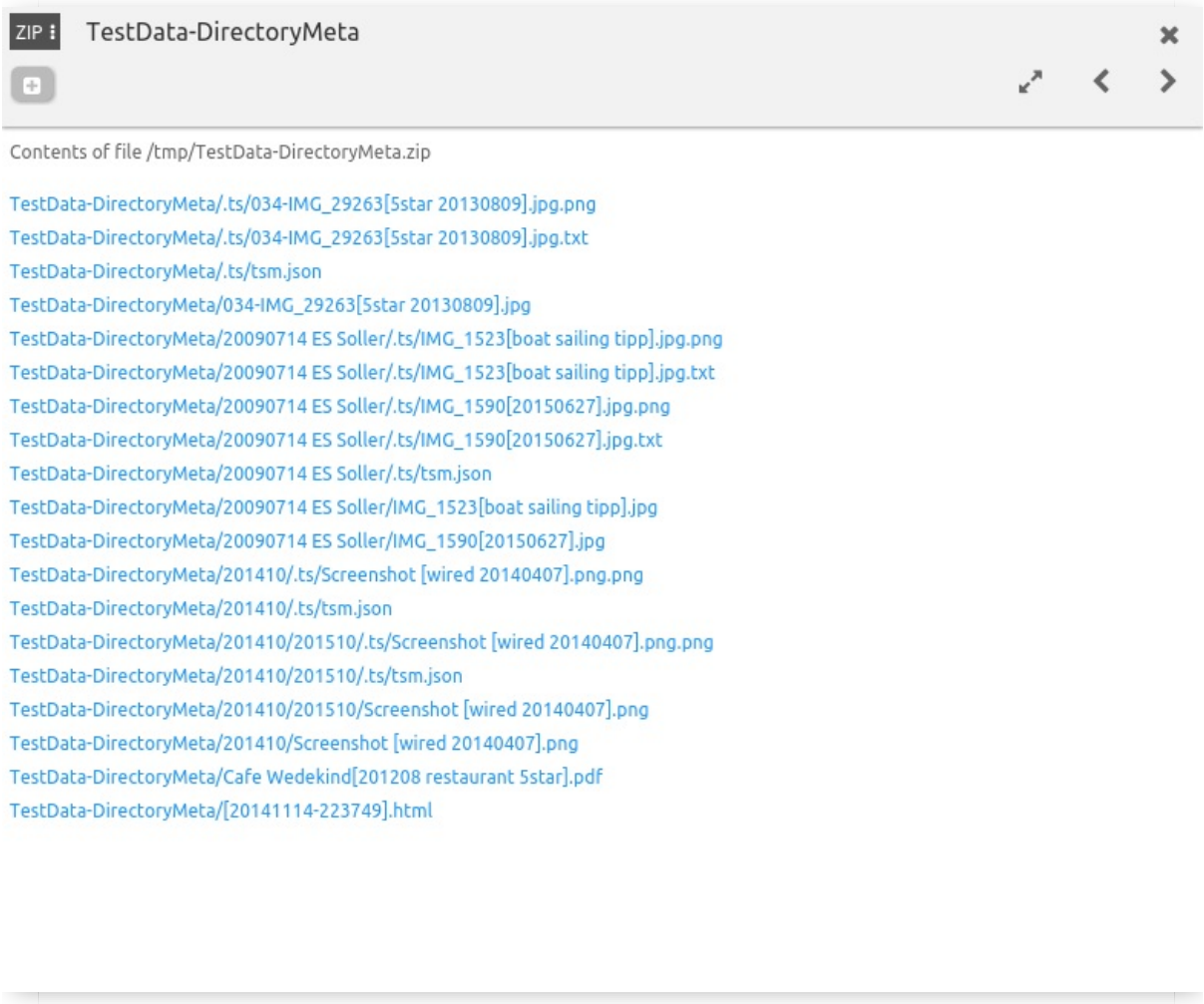
MIT

ZIP Preview for TagSpaces

A TagSpaces extension allowing you to preview the content of ZIP files.

Features

- Listing the content of ZIP files
- Previewing the content of the files in the ZIP container



Used libraries

This extension thankfully relays on the following great project(s):

- jszip

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/viewerZIP

Development

If you want to extend this extensions, please follow our general extension development guide

License

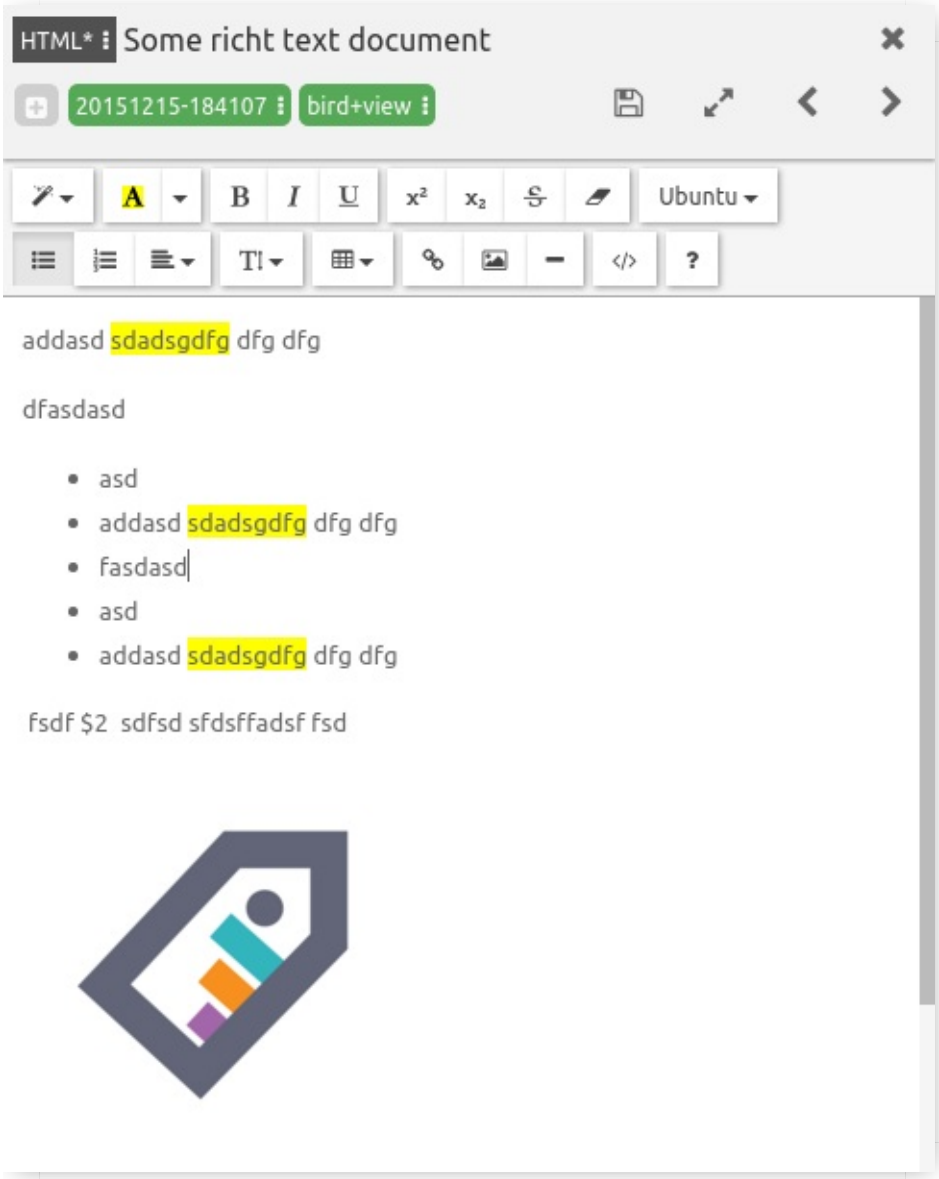
MIT

HTML Editor for TagSpaces

A TagSpaces extension allowing editing of HTML documents.

Features

- The standard feature set of the summernote javascript library
- Content cleaning and sanitizing
- Directly pasting images from the clipboard
- Saving all the images as dataurl in the one HTML file.



Used libraries

This extension thankfully relays on the following great libraries:

- summernote - a JavaScript library for HTML editing

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/editorHTML

Development

If you want to extend this extensions, please follow our general extension development guide

License

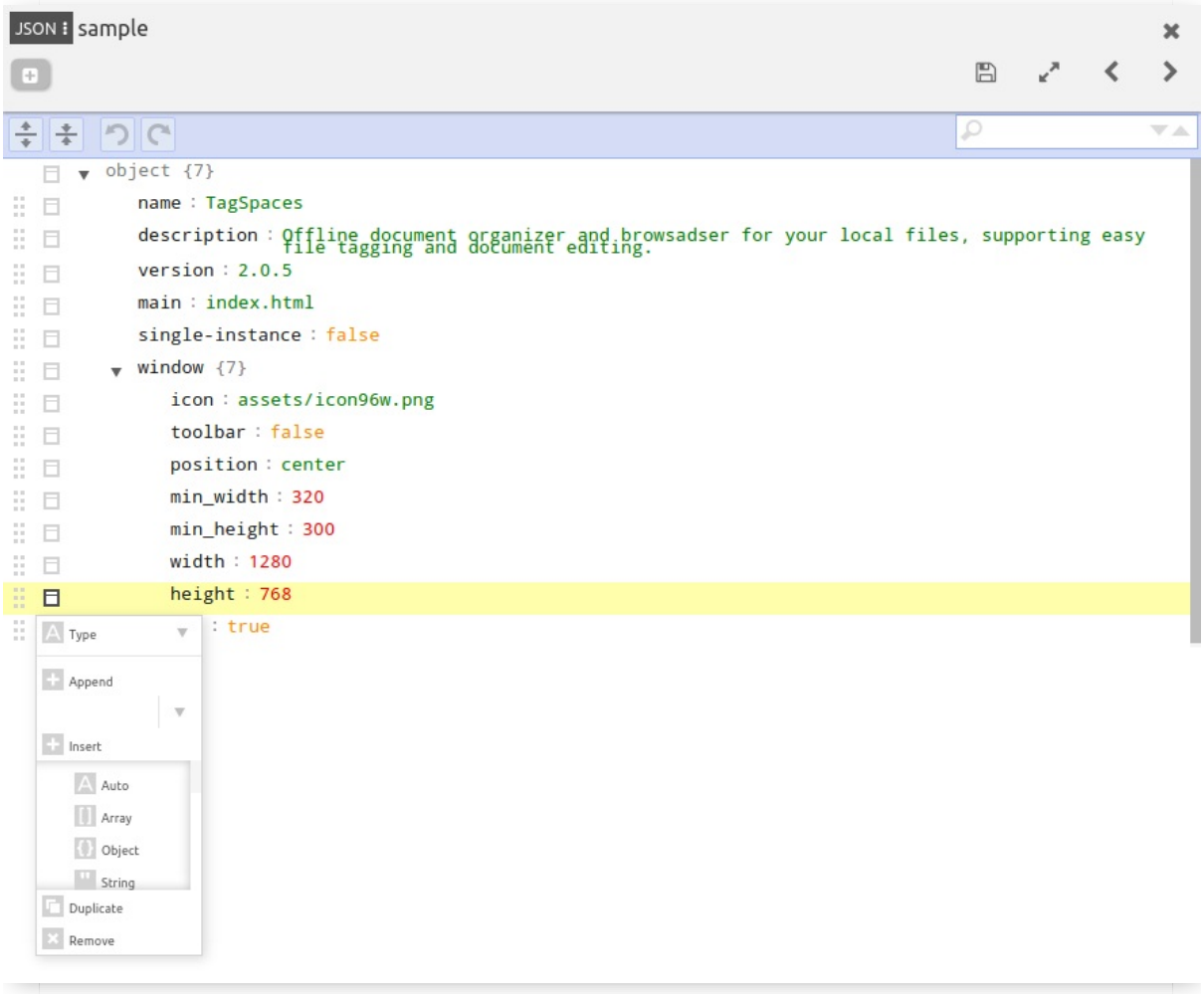
MIT

JSON Editor for TagSpaces

A TagSpaces extension allowing viewing and editing of JSON documents.

Features

- Open and edit JSON documents graphically



Used libraries

This extension thankfully relays on the following great libraries:

- jsoneditor

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/editorJSON

Development

If you want to extend this extensions, please follow our general extension development guide

License

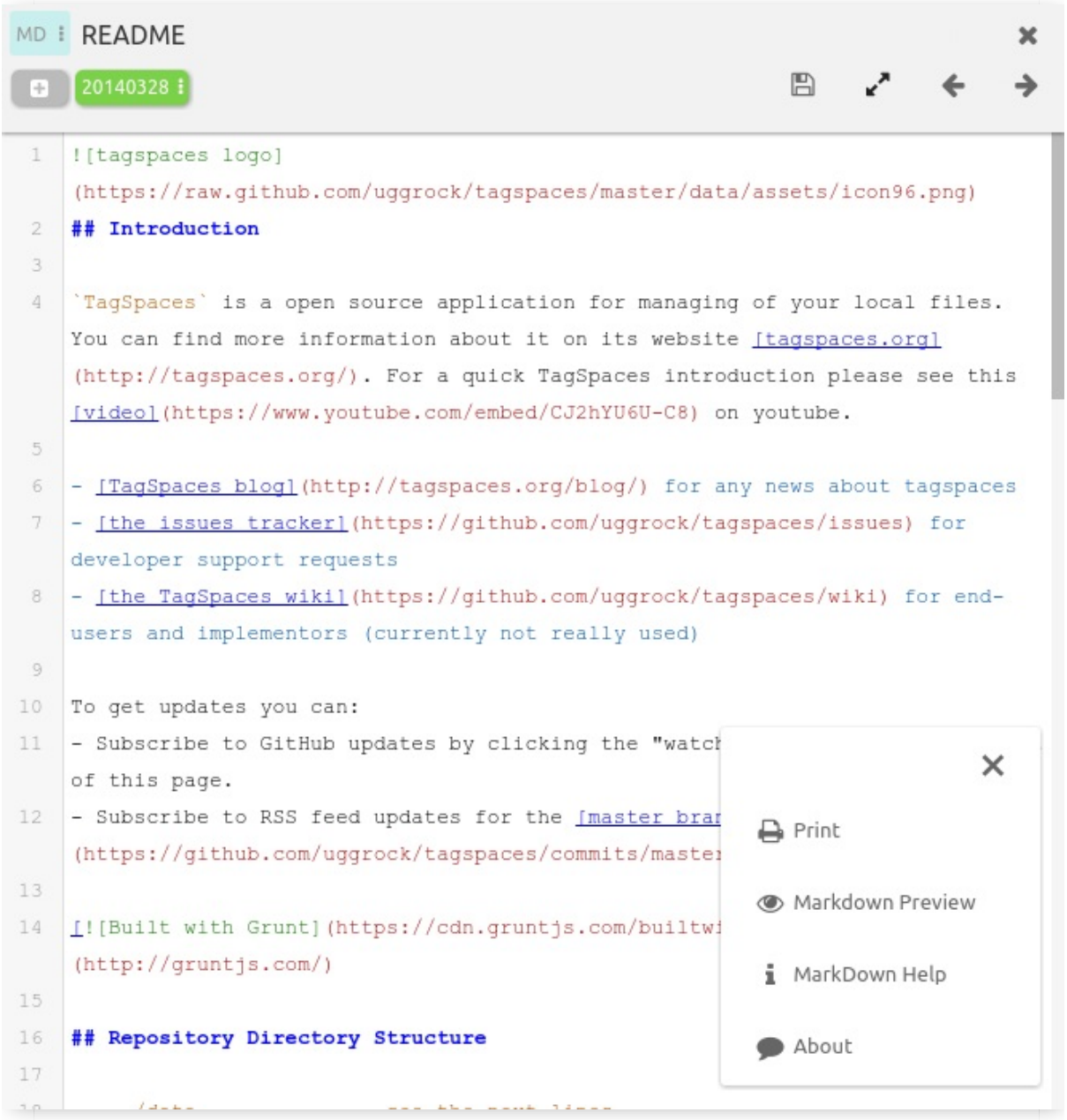
MIT

Text Editor for TagSpaces

A TagSpaces extension allowing editing of text based documents.

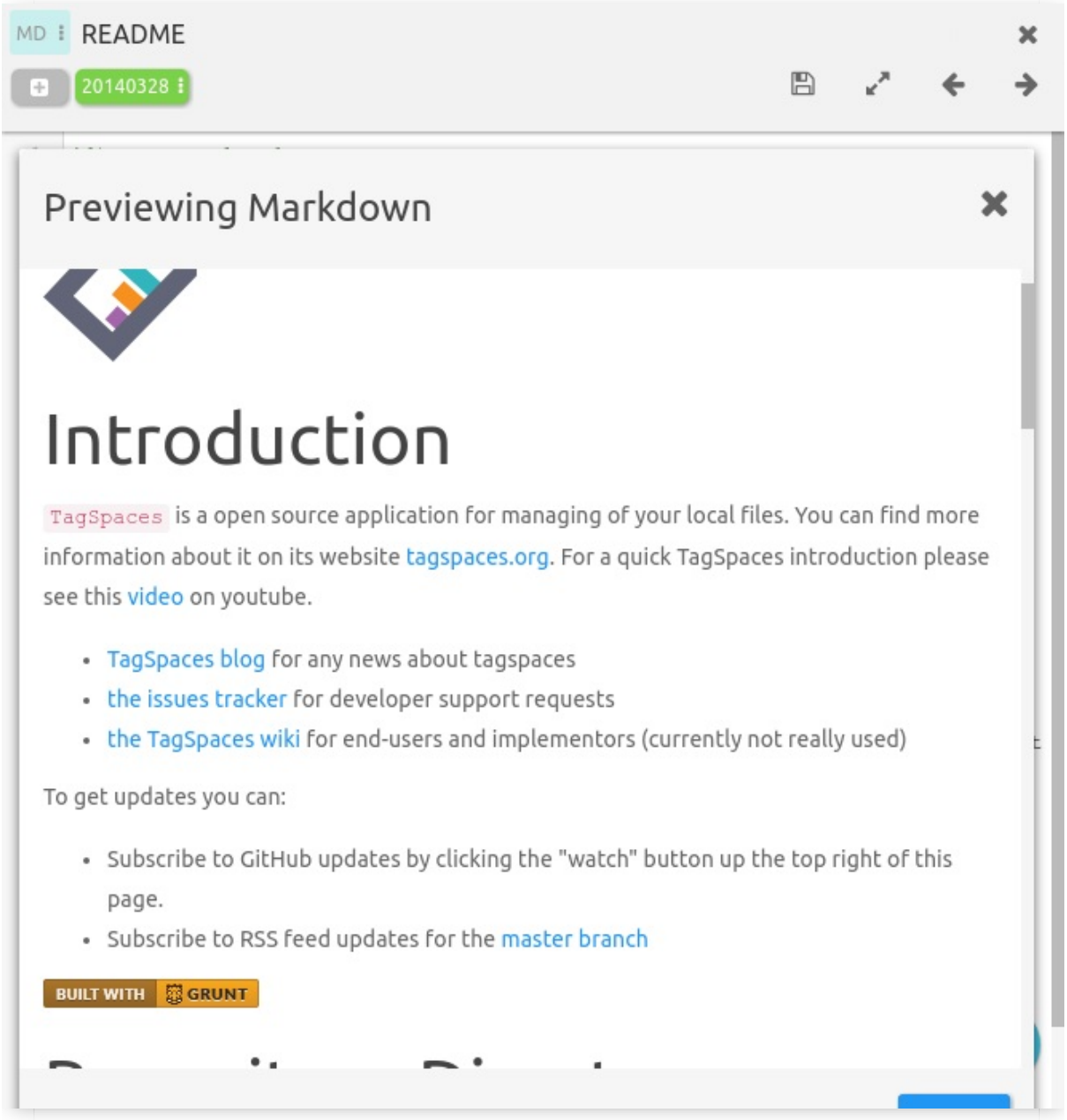
Features

- Opening and editing of text documents
- Syntax highlighting for the following file types: h, c, clj, coffee, coldfusion, cpp, cs, css, groovy, haxe, htm, html, java, js, jsm, json, latex, less, ly, ily, lua, markdown, md, mdown","mdwn, mkd, ml, mli, pl, php, powershell, py, rb, scad, scala, scss, sh, sql, svg, textile, txt, xml



Markdown preview

This text editor has an extended support for editing markdown files. It offers a preview of the currently edited markdown document and a short markdown syntax documentation. The markdown preview can be started from the menu item `Markdown Preview` of extensions main menu.



Used libraries

This extension thankfully relays on the following great libraries:

- codemirror

Installation

This extensions is packaged with any new version of TagSpaces

Source code

The source code of this extension is freely available on github.com/tagspaces/editorText

Development

If you want to extend this extensions, please follow our general extension development guide

License

MIT