Textastic for iOS Manual

Release 10.0

Table of Contents

Table of Contents

1 Introduction

Welcome to the documentation of Textastic, the powerful and fast code editor for iPad and iPhone.

Textastic supports **syntax highlighting** of over 80 programming and markup languages, remote connections (including (S)FTP, Dropbox, Google Drive, WebDAV, and SSH), search and replace (supporting regular expressions), web preview (including Markdown preview and a JavaScript console) and many more features to make coding on iOS easy and comfortable.

2 Managing Files

2.1 Local Files, iCloud, External Folders

Textastic offers two default locations in which you can store files: *Local Files* and *iCloud*. You can also add external folders using the *Add External Folder…* command (iPhone and iPad) or using drag and drop (iPad-only).

All these locations can be used to download files to and upload files from in the Remote File Transfer Screen.
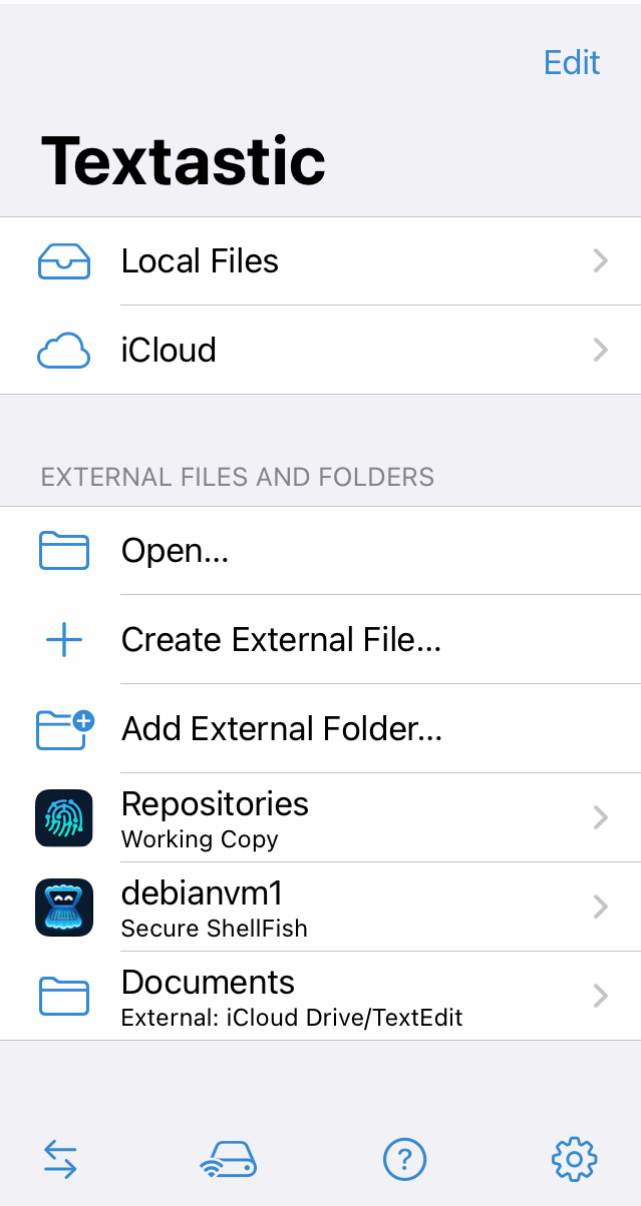
2.1.1 Local Files

Files saved in here are stored locally on your iOS device. They are not automatically synced to any cloud servers. Local files can also be accessed from the **Files** app in the "On My iPad/i-Phone" location under the folder called "Textastic".

You can also use **iTunes File Sharing** to add files to this location. Please have a look at this Apple support article for details.

2.1.2 iCloud

Files saved in here are stored in Textastic's **iCloud Drive** container. Changes you make in this folder are *automatically synced* to your other iOS, macOS, and Windows devices that are logged in to the same iCloud account.

You can find this folder in the **Files** app on iOS, in **Finder** on macOS and in **Explorer** on Windows:
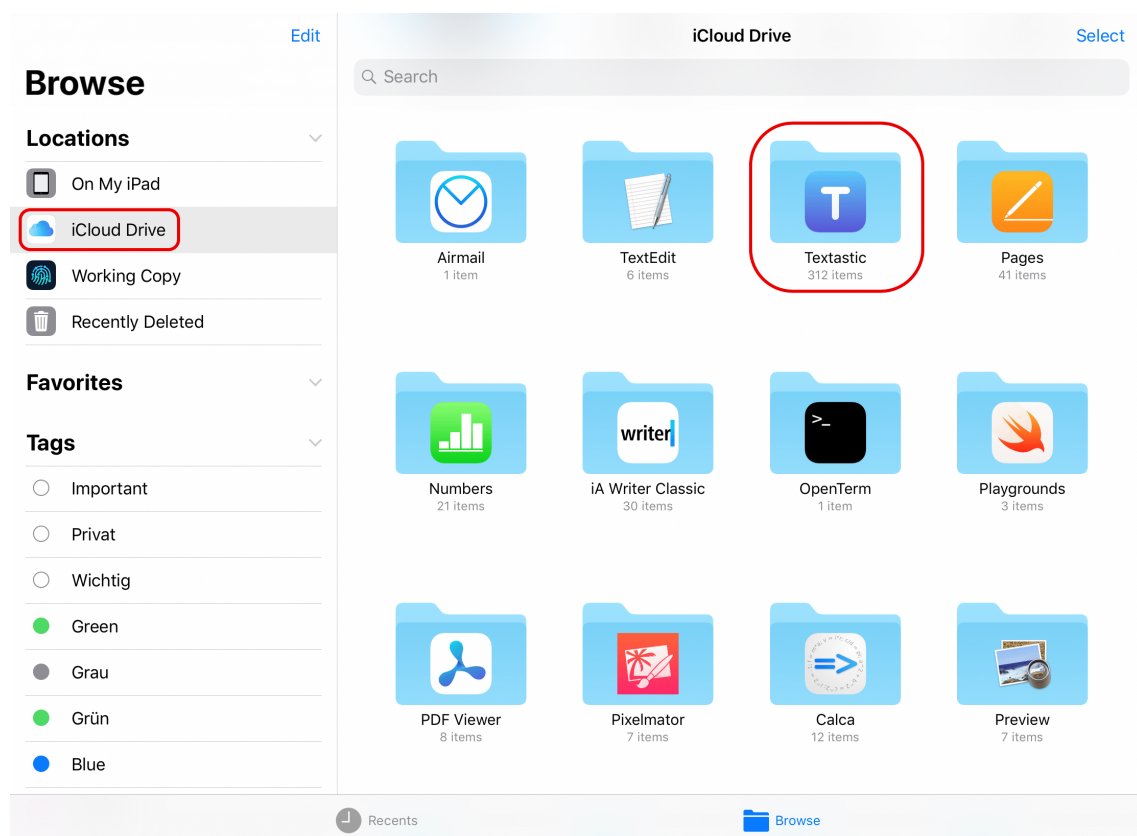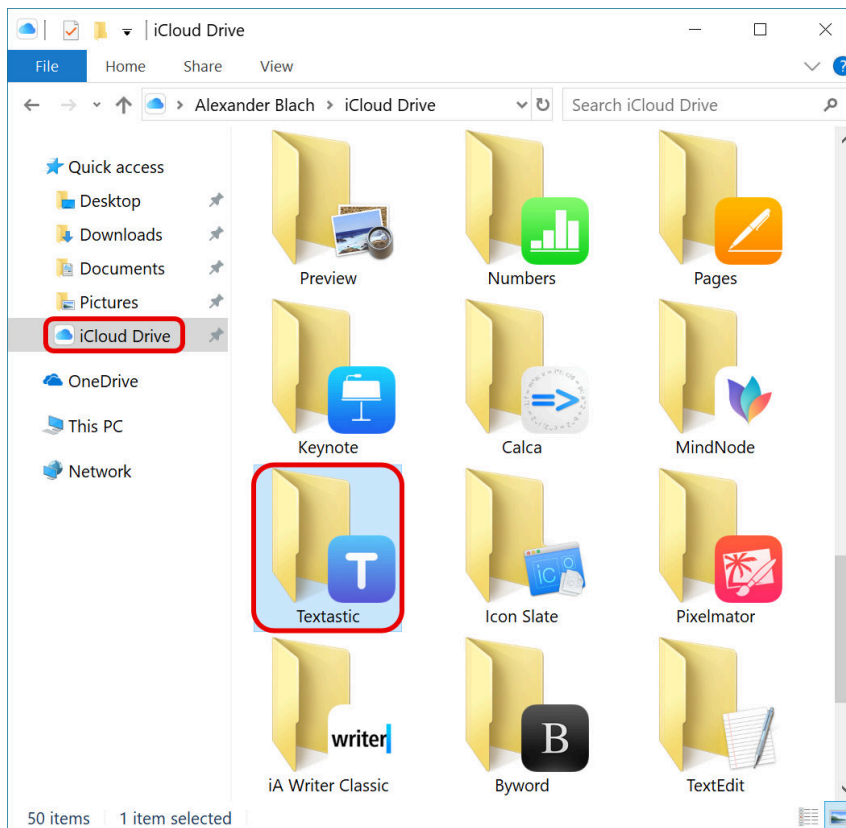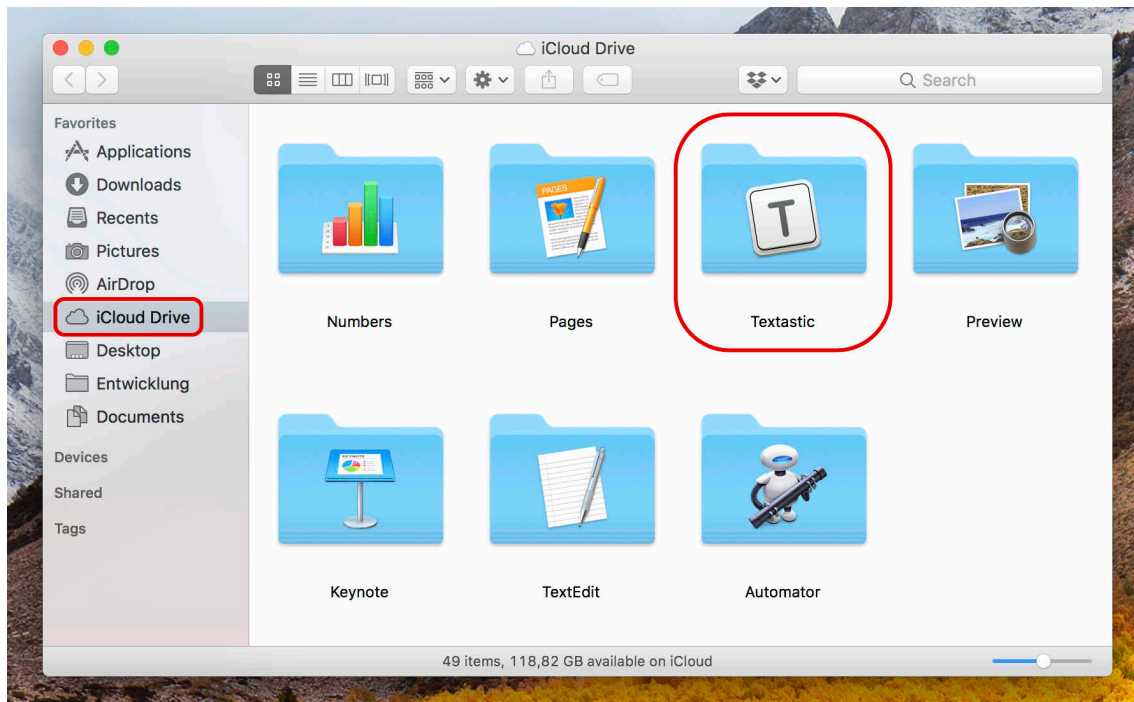
Note To make sure that Textastic's iCloud folder appears on all your devices, first create a file in the iCloud location in Textastic.

It is a rare issue, but for some users, it can take a while (up to 72 hours) until the folder appears in iCloud Drive on other devices.

2.1.3 External Folders

Textastic supports adding folders that are stored in another app through the use of the iOS document picker or Drag & Drop. This works with folders stored locally on the device in another app that has a file provider and is primarily used to access Git repositories from apps like Working Copy.

You can read more about this at External Files and Folders and Git Client Working Copy.

2.1.4 Open Files and Folders in the Files App

Invoke the context menu of any file, folder, or location and choose *Show in Files App* to open the folder in the Files app. For files, this will open the containing folder.
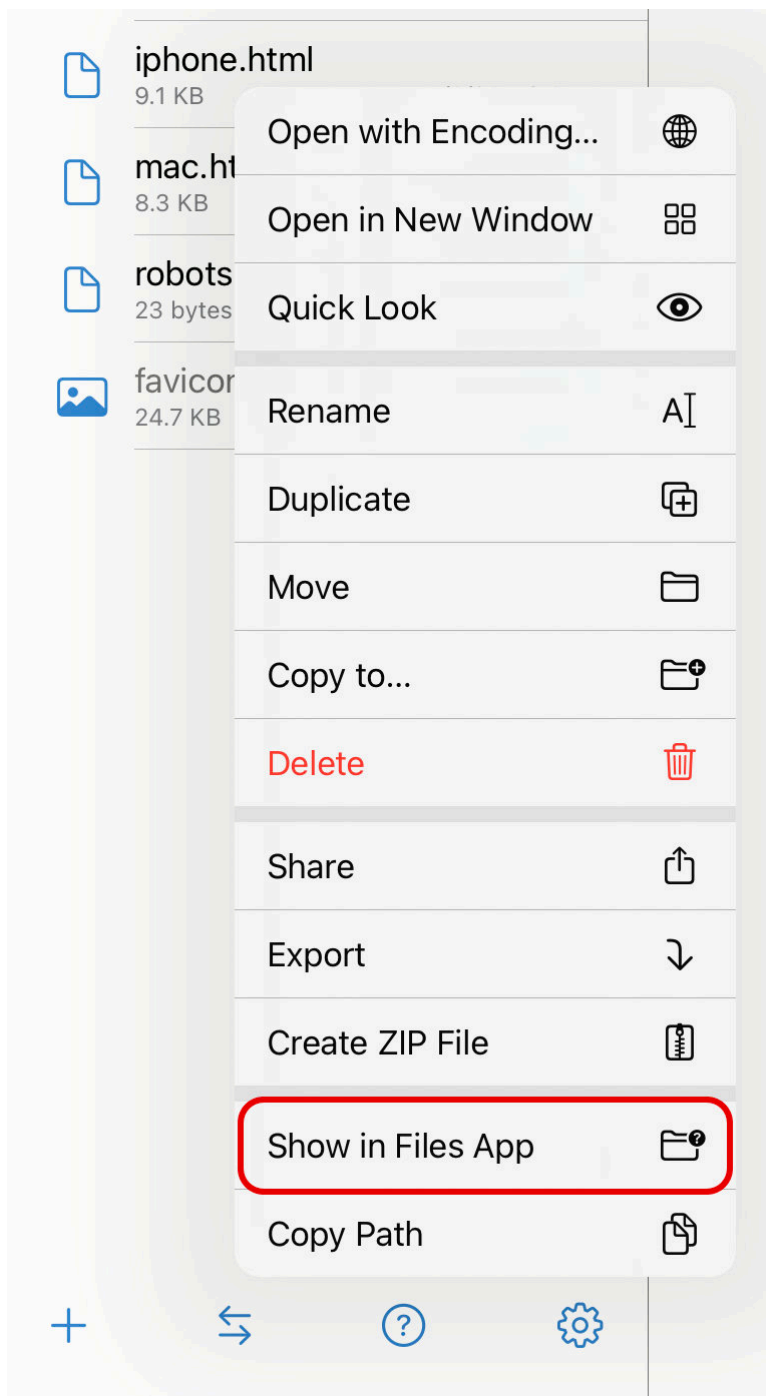
2.2 Adding New Files, Folders, and Images

2.2.1 Creating Files and Folders

To create a new file or folder, you first need to navigate to the directory in which you want to create it. This can be a folder in "Local Files", "iCloud" or in an external folder (see Local Files, iCloud, External Folders).

Then tap on the + button in the bottom toolbar.

In the *File or Folder* dialog you can select whether you want to add a *File* or a *Folder*, or whether to *Import* files from the Files app, or images from the photo library.

Filename / Folder Name

Enter the filename or name of the folder you want to create. Make sure to include a file extension (like ".html") when creating a new file.

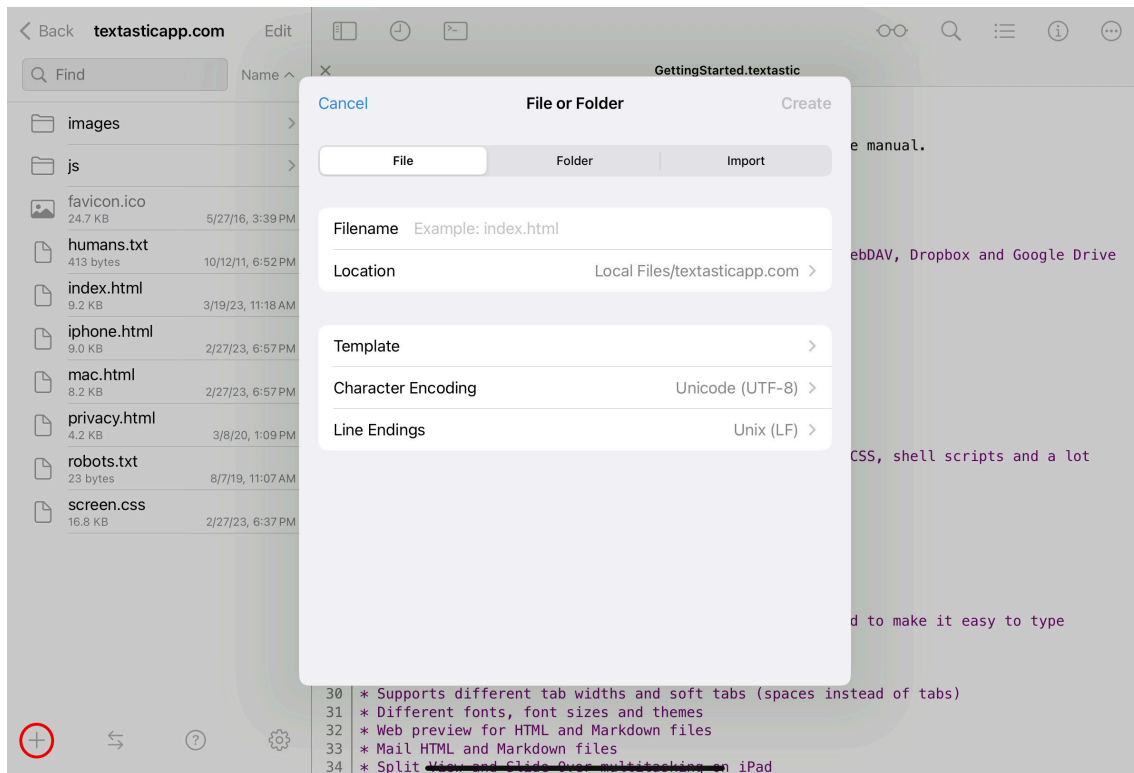**Location**

Here you can choose the location where you want to create the new file or folder.

Template

Optionally choose a template to use when creating a new file. There are built-in templates for HTML and XML files, but you can also Section 7.1.4.

Character Encoding

Select the text encoding of the new file. The default is UTF-8.

Line Endings

You can choose Unix (LF), Windows (CRLF), or Mac (CR) line endings when creating new files. The default is Unix (LF).

Please note that Mac (CR) line endings are uncommon: use Unix (LF) for files intended for modern macOS.

Character encodings and line endings you choose here are remembered for the next time you want to create a file, so you typically only need to set that once. UTF-8 and Unix line endings are a good choice for most.

Important If you create a file, make sure to enter a *Filename* **including the file extension**, since the extension is used to determine the correct syntax definition. If you choose a template, Textastic automatically adds the correct file extension to the file name.

Tap *Create* to create the file or folder.

2.2.2 Importing Files and Images
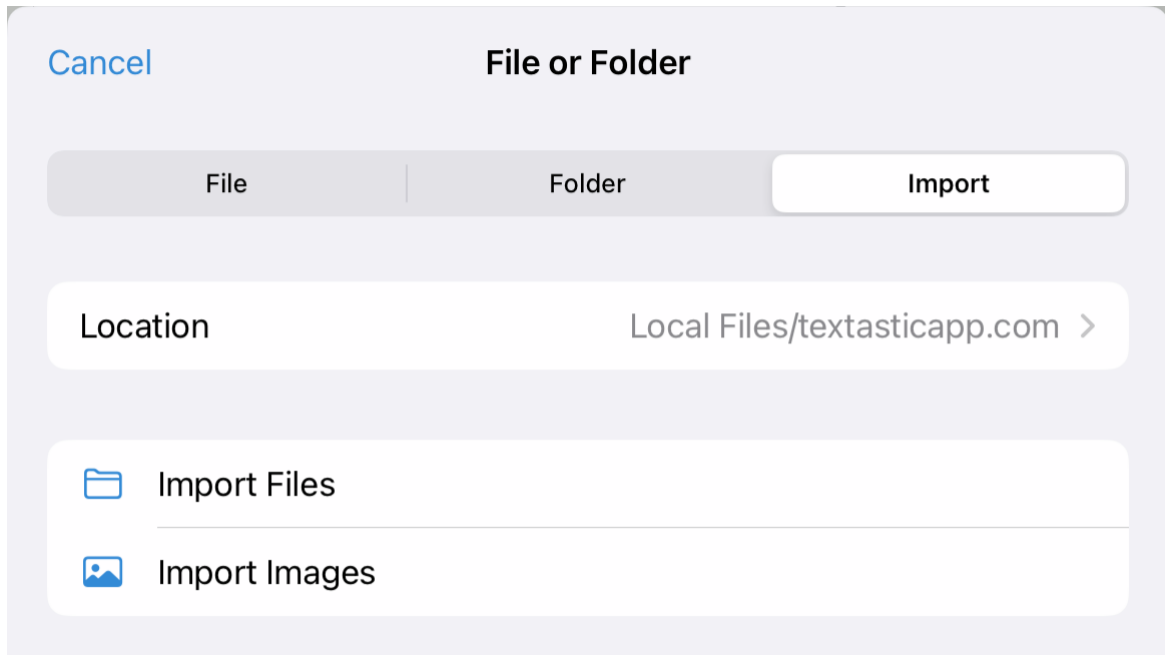
In the "File or Folder" dialog, switch to *Import*.

Tap *Import Files* to import files from the `Files` app. Imported files are not opened in place, but rather copied to the location specified in the dialog.

Tap *Import Images* to import images from the photo library in the `Photos` app.

Hint If you want to open files in place instead of importing/copying them to an existing folder, have a look at External Files and Folders.

2.2.3 Add Button Context Menu
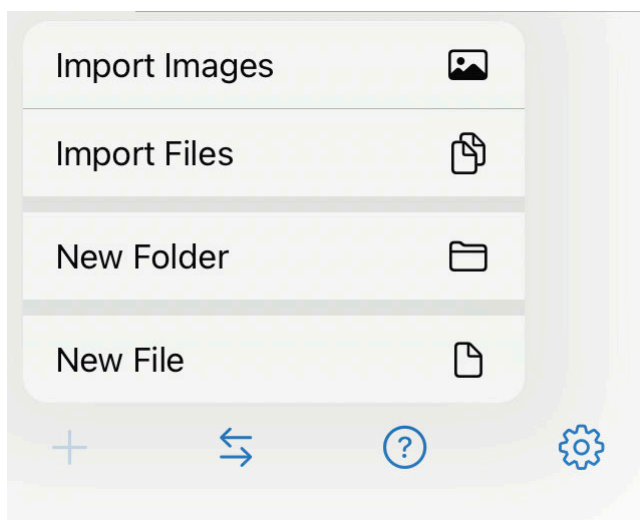
Alternatively, tap and hold the + button to show its context menu and directly add files or folders or import files or images.

2.2.4 Creating External Files

To create a new file in a location outside of *Local Files*, *iCloud* or external folders, use the command *Create External File…* (see Local Files, iCloud, External Folders). This shows the same options as seen in the screenshot above, but allows you to pick any location in the `Files` app.

2.2.5 Importing Images using Drag and Drop

On iPad you can also use drag and drop to import images from the `Photos` app into Textastic.

2.3 Sorting File Lists

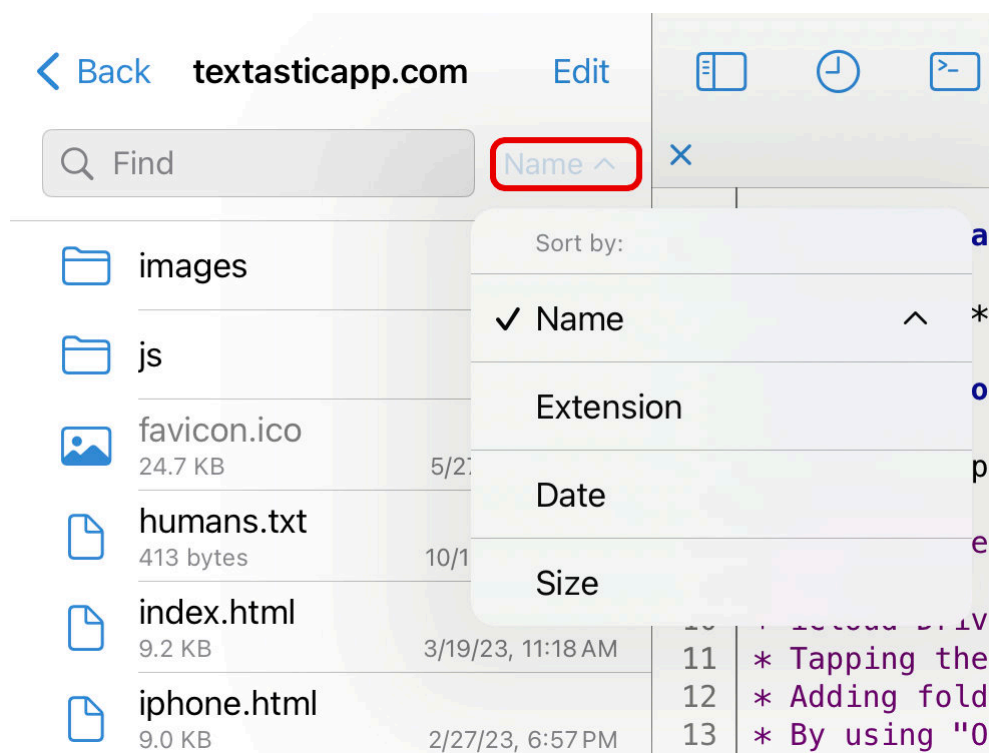Tap on the button next to the search field on the top of the file list to change the sort order of file lists (default is *By Name*). You can sort by name, file extension, date, and size.

A small arrow on the right side of the button label indicates the sort order. To **toggle the order between ascending and descending**, tap on the button again and select the same sort criteria. For example, if you select *Sort by: Name* for the first time, the file list will be sorted in ascending order. When you then select it again, it will be sorted in descending order.

2.4 Finding Files by Name

Use the fuzzy search for file names to find and open a file by typing characters that are in the name of the file or path. This searches the current folder and its subfolders.
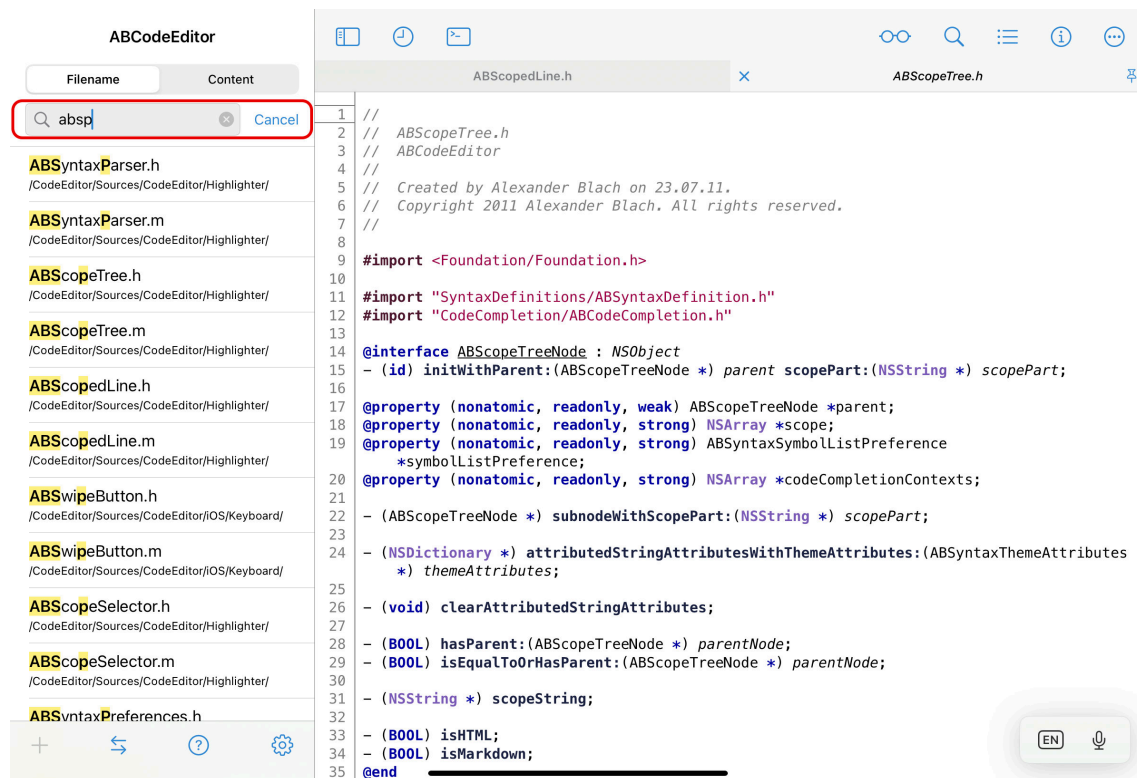
A sophisticated scoring algorithm orders the search results, so you'll find the most likely match on the top of the list. Consecutive character matches, acronyms, and matching file extensions get a higher score.

If you are looking for a file named `ABSyntaxParser.h`, try searching for the acronym "absp" or for "sp.h" and it will most likely appear at the top of the list.

2.5 Find in Files

You can **search the contents** of all text files in the current folder and its subfolders for matching text or regular expressions.

Tap the search field above a file listing and change the search mode from *Filename* to *Content*.

textasticapp.com
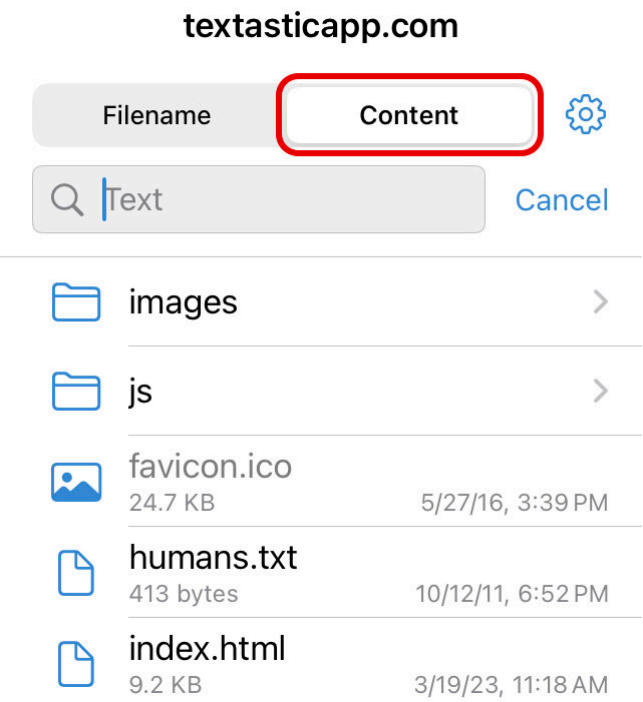
See also

You can also search and replace in the currently active file.

2.5.1 Settings

Tap the *Search Settings* button ("gear" icon) to open the file content search settings screen.

Here you can choose whether to match text or a regular expression. You can also specify whether the search should be case sensitive, diacritic sensitive, and whether to search for whole words only:
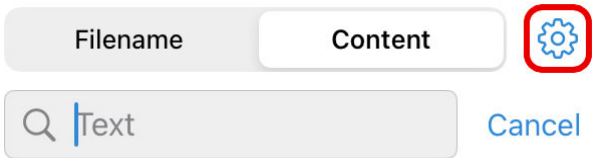
< Back   **Search File Content**

MATCH

| ab | Text | ✓ |

| .* | Regular Expression |

OPTIONS

| Aa | Case Sensitive |

| öo | Diacritic Sensitive |

| ab | Whole Words |

Text

Search for a simple text string. You can also search for "\n" or "\r" to find newline charac-
ters and "\t" to find tab characters. Enter "\\" to search for a backslash.

Regular Expression

Enable regular expression search. Textastic uses the ICU regular expression syntax.

Case Sensitive

By default, search is case-insensitive. Enable this setting to make search case sensitive and
only find instances with matching case.

Diacritic Sensitive

By default, search ignores diacritic marks: for example, 'ö' is equal to 'o' and 'é' is equal to
'e'. Enable this setting to only find letters with matching diacritic marks.

Note this setting is not shown when using regular expression search. Regular expression
search is always diacritic sensitive.

Whole Words

If you enable this setting, the search will only find whole words. For example, a search for
"word" will find "word" but **not** "wording" or "password".

2.5.2 Performing the Search

Enter your search term in the search text field and start the search by pressing the `Return` key.
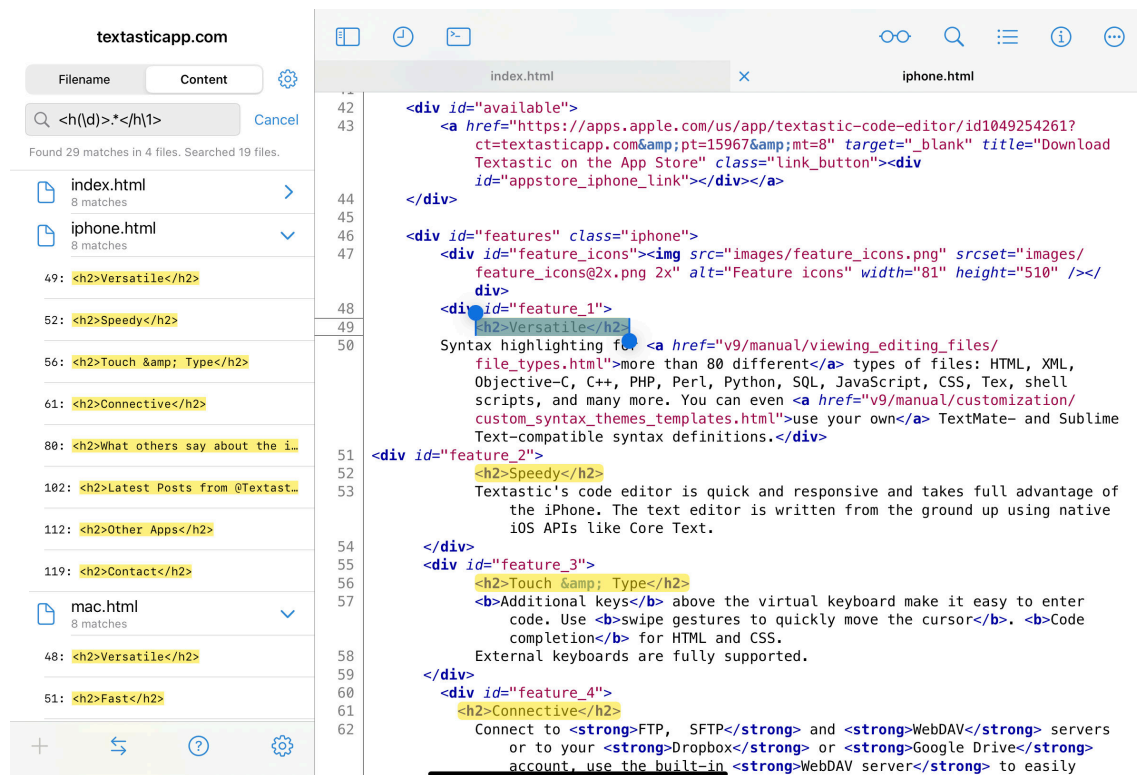
After a folder search you will see the search results grouped by file and ordered by filename. Matches found in files in the current folder are shown before search results in subfolders.

2.5.3 Navigating Search Results

Tap a search result to open the file and jump to the search result.

As soon as the search is finished, you can collapse and expand the search results by tapping on a file name.

Use the context menu to expand or collapse all search results:
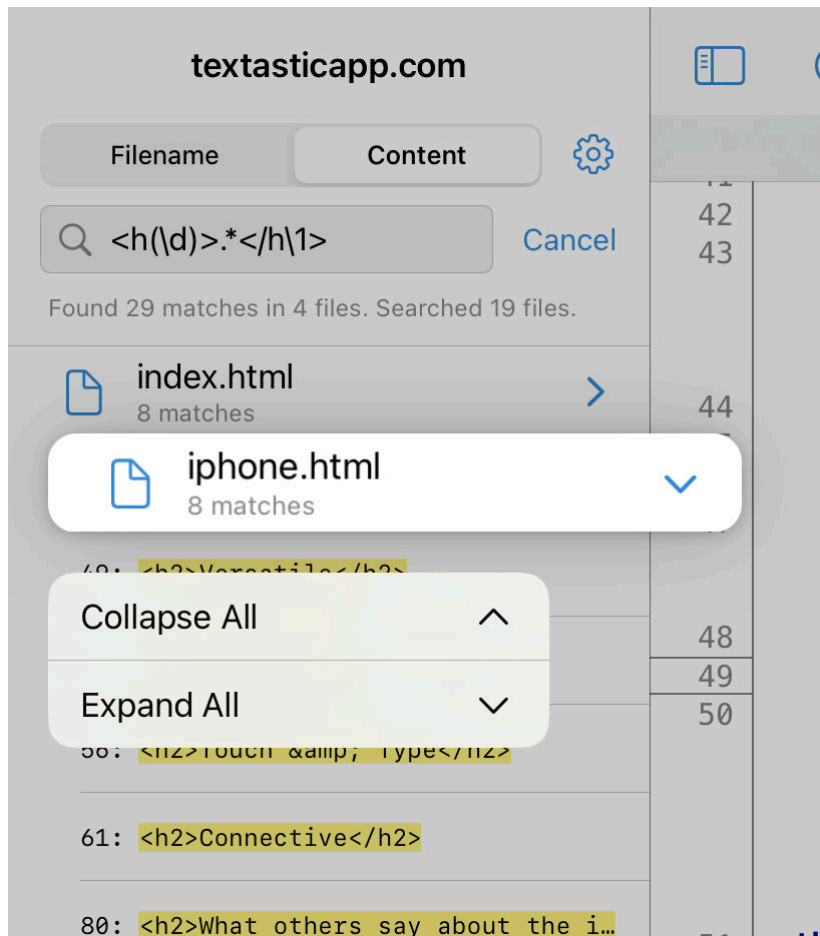
Note Searching multiple files works in Local Files, iCloud and external folders (such as other folders in iCloud Drive or Git repositories in Working Copy).

2.5.4 Applying the Same Search Term and Settings to Single File Search

If you try to search in the currently active file while a search in multiple files is active, Textastic adds a *Search for "…"* button to the search text field.

Tap this button to quickly use the same search term and search settings in the single file search as in the multiple file search:
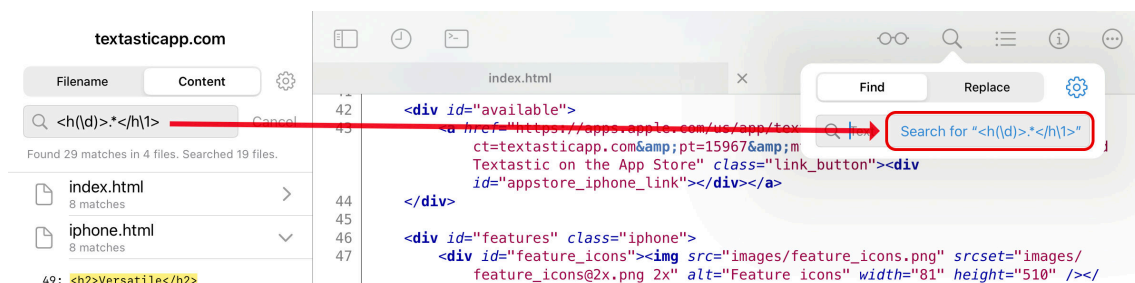
2.6 Moving, Copying, and Renaming Files and Folders

The *Edit* button above file lists allows you to move, copy, delete, share, and rename local files and folders. Tap on the button to switch to edit mode:
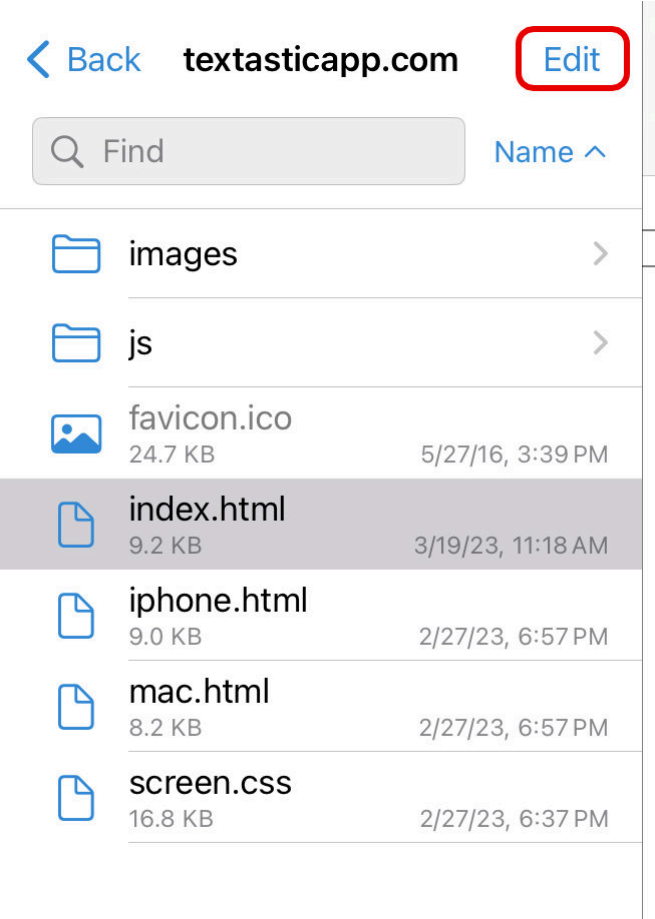
You can select files and folders by tapping on them.
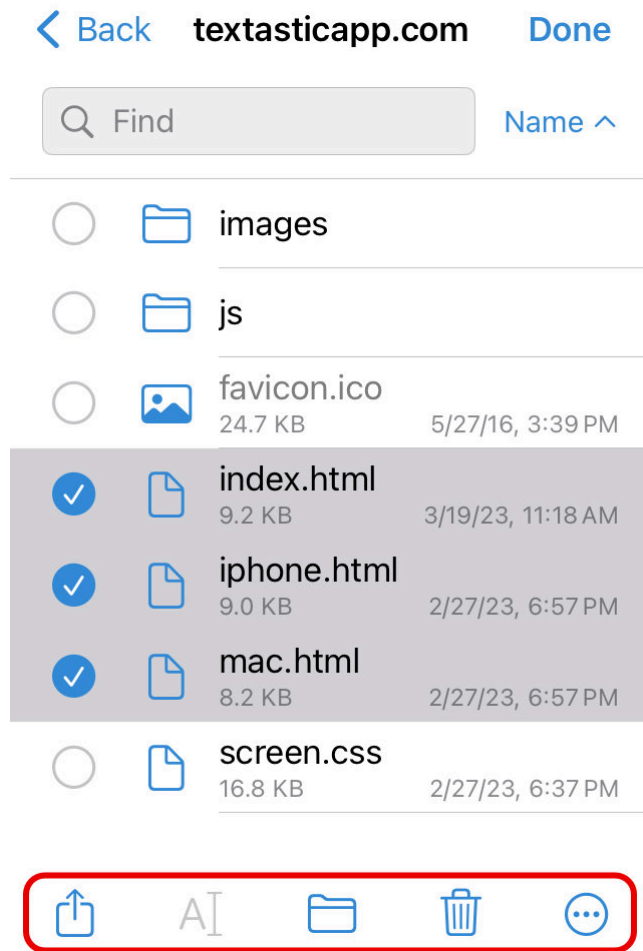
Using the buttons on the bottom you can then *Share*, *Rename*, *Move*, and *Delete* the selected files and folders.

Tap on the *More* button to show the following additional actions: *Duplicate*, *Copy To…*, *Export*, and *Create ZIP File*:
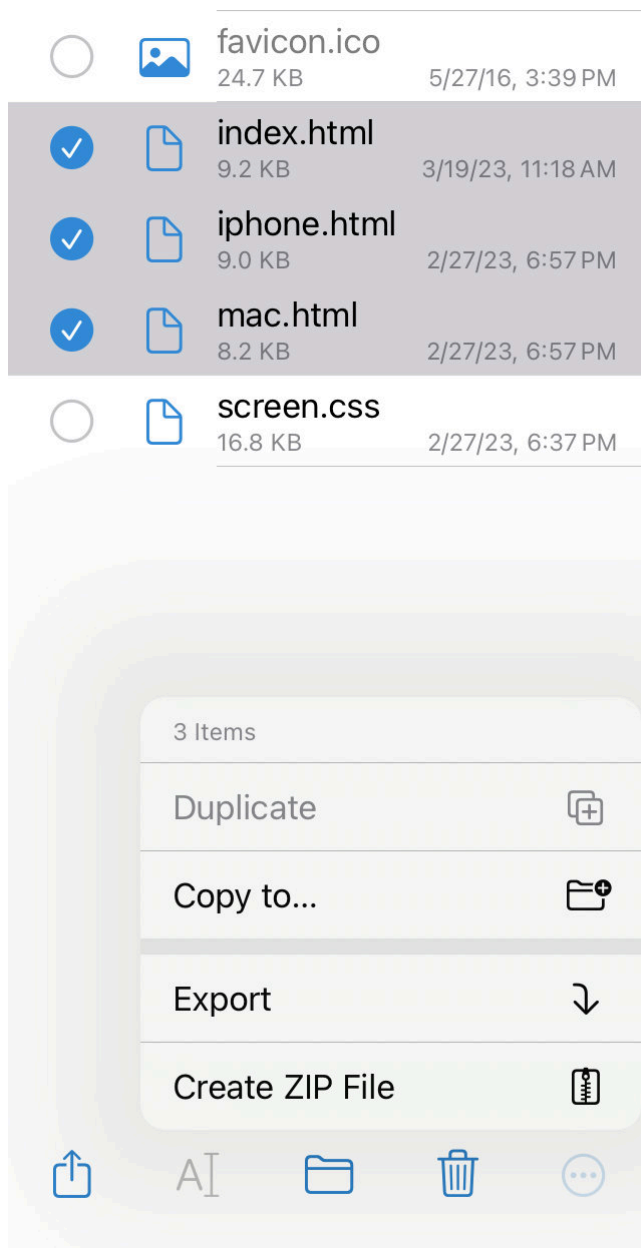
Those buttons and actions become active as soon as at least one file is selected. *Rename* and *Duplicate* are only available when exactly one file or folder is selected.

2.6.1 Moving Files

If you tap on the Move button, Textastic will bring up a list of your folders. Choose the folder where you want to move the selected files or folders.

< Back   **Select Folder**

📥 Local Files   ∨

📁 #Textastic   ›

📁 ScreenshotData   ›

📁 logs   ›

📁 projects

📁 ssh

📁 textasticapp.com   ∨

📁 images

📁 js

☁ iCloud   ›

Repositories   ∨

📁 Textastic   ›

📁 WeatherStatus   ›

📁 cpython   ›

📁 three.js   ›

debianvm1   ›

📁 Documents
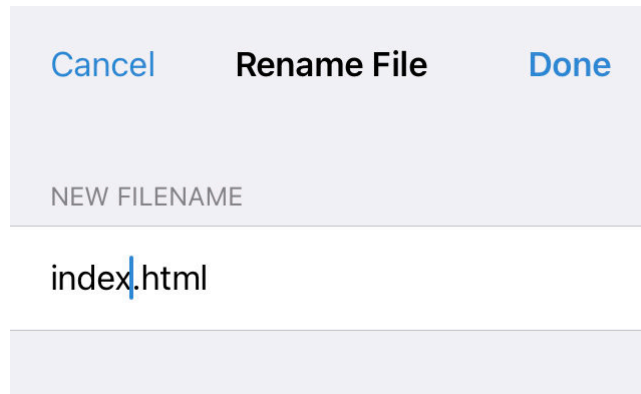
Move

You can expand folders by tapping on them.

Once the target folder is selected, tap on the *Move* button in the lower right corner.

2.6.2 Renaming and Copying Files

Tap the buttons in the bottom toolbar to *Rename* or *Duplicate* a single file or folder.

After you choose to duplicate or rename a file or folder, Textastic will ask you for a new name.

Cancel   **Rename File**   Done

NEW FILENAME

index.html

Use the *Copy To…* command to copy multiple files or folders to another folder.

2.6.3 Drag and Drop Files

You can also use **drag and drop** to **move** files and folders to another folder within Textastic: drag files with one finger and use another finger to navigate into directories.

On iPad, you can also drag files and folders from other apps into a file list in Textastic to **copy** them.

Dragging Multiple Files

- Touch and hold the first file or folder, drag it slightly, and continue holding it.
- While still holding the first item, tap additional files with another finger. A badge indicates the number of selected items.
- Drag all of the items together.

2.7 Recently Opened Files

You can show a list of recently opened files by tapping on the *Recent Files* button.
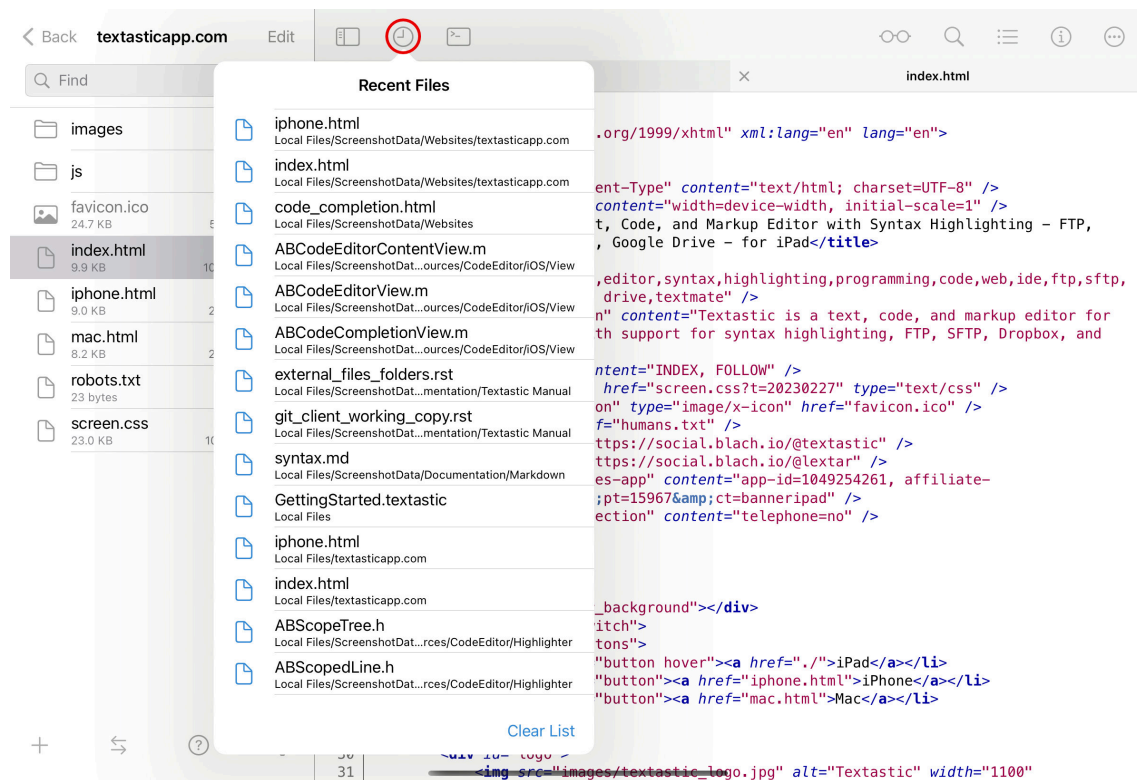
Use the *Clear List* button to remove all files from the list or swipe on an entry in the list to the left to delete only one of the files from the list.

3 Connecting to Remote Servers

3.1 Remote File Transfer Screen

Tap on the *File Transfer* (two arrows) button in the bottom toolbar to open the File Transfer screen.
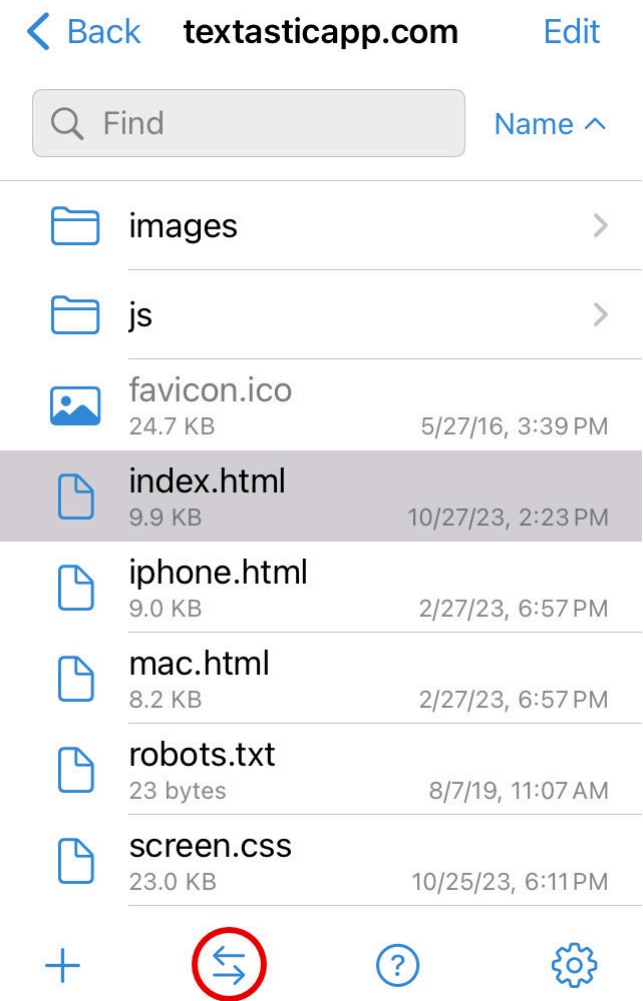
You can use it to download files from remote servers and services and to upload files.

3.1.1 Local Files on the Left, Remote Files on the Right

Inside the File Transfer screen, you'll see two panes: your local files on the left, and remote connections and files on the right.

On the left side, you can navigate to any folder in Local Files, iCloud or external folders.

On the right side, you can create connections to SFTP, FTPS, FTP, and WebDAV servers and to Dropbox and Google Drive accounts. By tapping on a connection, you can connect to the remote server or service and navigate to any folder within it.

3.2 SFTP, FTPS, and FTP connections

3.2.1 Add an (S)FTP Connection

After opening the File Transfer screen, tap on the + button and choose *(S)FTP* to add a new connection.

Then you can configure a new Section 3.2.2, Section 3.2.3 or Section 3.2.4 connection.

3.2.2 Configure an FTP (Unencrypted) Connection

Choose the *FTP (Unencrypted) Protocol*.

ansfer

Cancel	(S)FTP Connection	Done

Title	Optional

Protocol	FTP (Unencrypted)	>

Host	textasticapp.com

Port	21

User	user

Password	●●●●●●●●●●●●●●●●●●●

Ask for Password	

Path	~/

Encoding	Unicode (UTF-8) >

Connections	3

Debug Log	

FTP

Connection Mode	Passive (EPSV/PASV) >

Show Hidden Files	

Configure the connection by setting the *Host* name (this can be a domain name like "example.-

com"or an IP address), a *User* name and a *Password*.

If you check *Ask for Password*, you do not need to enter a password. Instead, Textastic will ask for the password when connecting to the server.

Important Make sure that the host name does not include a protocol like "ftp://" or a slash "/" after the host name or the connection won't work.

You can also add a custom *Title* that will appear in the connection list and choose another port than the standard port 21.

If you enter a *Path*, Textastic will use this as the root path when connecting to the server.

Choose the correct character *Encoding* you want to use for this connection. This setting depends on your server's configuration.

By default, Textastic uses up to three concurrent *Connections* when uploading or downloading multiple files. If your FTP server only allows a single connection, you can set this to "1" instead of "3".

If you encounter connection problems with the standard *Passive (EPSV/PASV) Connection Mode*, try to use *Active (EPRT/PORT)*` instead.

Enable the *Show Hidden Files* setting if you want to see hidden remote files like `.htaccess`.

3.2.3 Configure an FTPS (FTP over SSL) or FTPES Connection

For FTPS / FTP over SSL connections, most settings are the same as for regular FTP connections (see above). Notice the **different default port 990**.

You can also choose *FTPES (FTP over Explicit Auth SSL)* as the protocol which **defaults to port 21**.

FTPS adds a *Verify SSL Certificate*. Disable this if you use a **self-signed certificate** on the server.

3.2.4 Configure an SFTP (SSH) Connection

SFTP (SSH File Transfer Protocol) is a subsystem of SSH (Secure Shell). You can use this connection type to securely transfer files between your server and Textastic. For more information have a look at SSH File Transfer Protocol (Wikipedia)

The **default port** for SFTP connections is **22**.

Textastic usually switches to the root directory after establishing an SFTP connection. If you want to start at the **user's home directory** instead, enter ~/ in the *Path* field.

You can either use user/password authentication or public key authentication:

User/Password authentication

Enter your user name and password as you would for a regular FTP connection, but *disable* the *Public Key Auth.* setting.

Public Key Authentication

Enter the **user name** and *enable* the *Public Key Auth.* setting. Now you can enter the path to the private key file.

Important The private key file must be stored somewhere in Local Files.

For this to work, you need to transfer the private key file for your SSH account into Textastic. You can do this for example by using Dropbox, Google Drive or the built-in WebDAV server (via Wi-Fi).

You can also use the **Files** app and move the private key file into a folder in "On my iPad/i-Phone" -> Textastic or use **iTunes File Sharing**.

The standard settings expect a private key file named id_rsa in a subfolder called ssh in Local Files in Textastic (see screenshot).

If your private key file is protected with a **passphrase**, Textastic uses the **password** setting to retrieve the private key.

Note Textastic supports RSA, DSA, ECDSA, and Ed25519 keys.

Key File Format

Textastic expects the key file to be in the format which is used by the **OpenSSH utility ssh-keygen** as well as the **OpenSSL** command line tool. It supports the new OpenSSH private key format as well as the PEM format. When using **ssh-keygen**, the file is usually called id_rsa while with **openssl**, it is usually called private.pem.

If you instead have a key in PuTTY's format (.ppk), you can use PuTTYgen to convert it to OpenSSH/PEM format:

Visit the PuTTY website and download puttygen.exe (on your PC). Open it, and using the *File* menu, load your .ppk file that you already use, and type in your passphrase. Now go to the *Conversions* menu and export the key as an OpenSSH key. You can now use the converted key with Textastic.

3.2.5 Debugging Connection Problems

If you encounter connection problems, you can enable the *Debug Log* setting. This will create detailed log files in the "logs" directory in Textastic. Note that **this logs all traffic for this connection**, so only enable this for debugging purposes and make sure to disable it again as soon as possible.

3.2.6 Save Connection

Tap *Done* to save this connection and return to the list of remote connections.
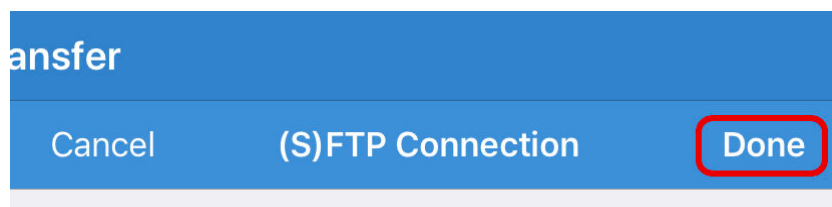
3.2.7 Connect

You can now tap on the newly created connection to connect to the remote server.
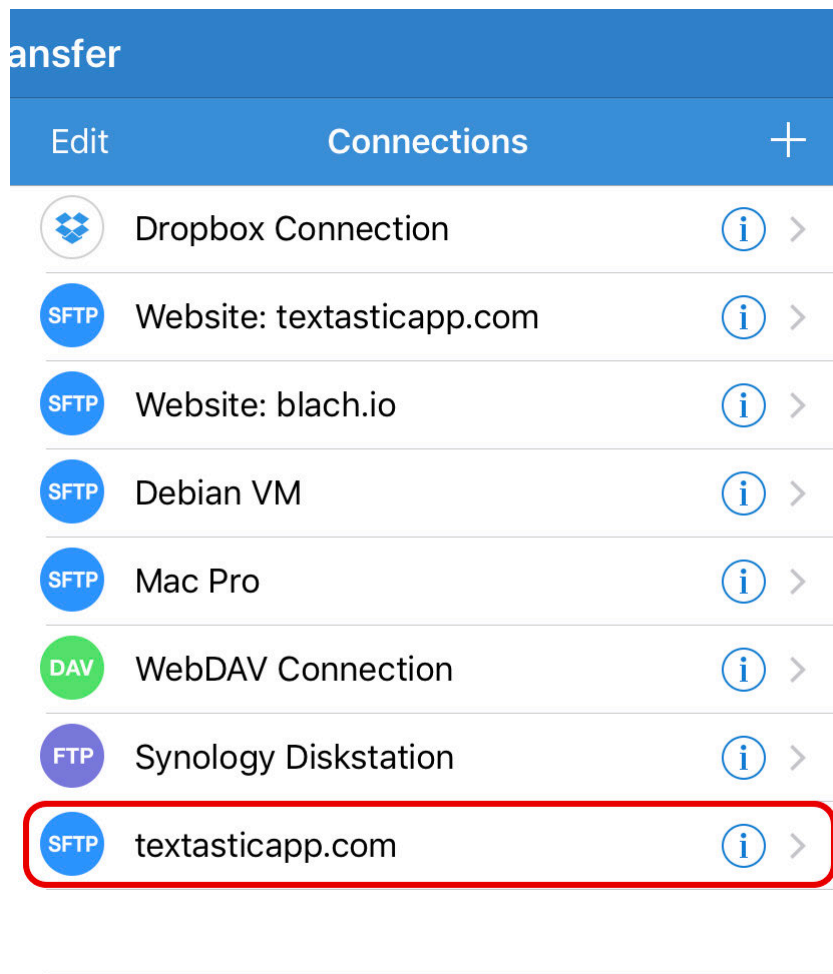
3.2.8 Change Connection Settings

You can change the settings of a connection by tapping on the *i* button on the right side of a connection.
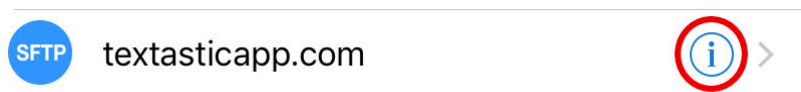
3.3 Dropbox Connections

3.3.1 Add a Dropbox Connection

After opening the File Transfer screen, tap on the + button and choose *Dropbox Connection* to add a new connection.
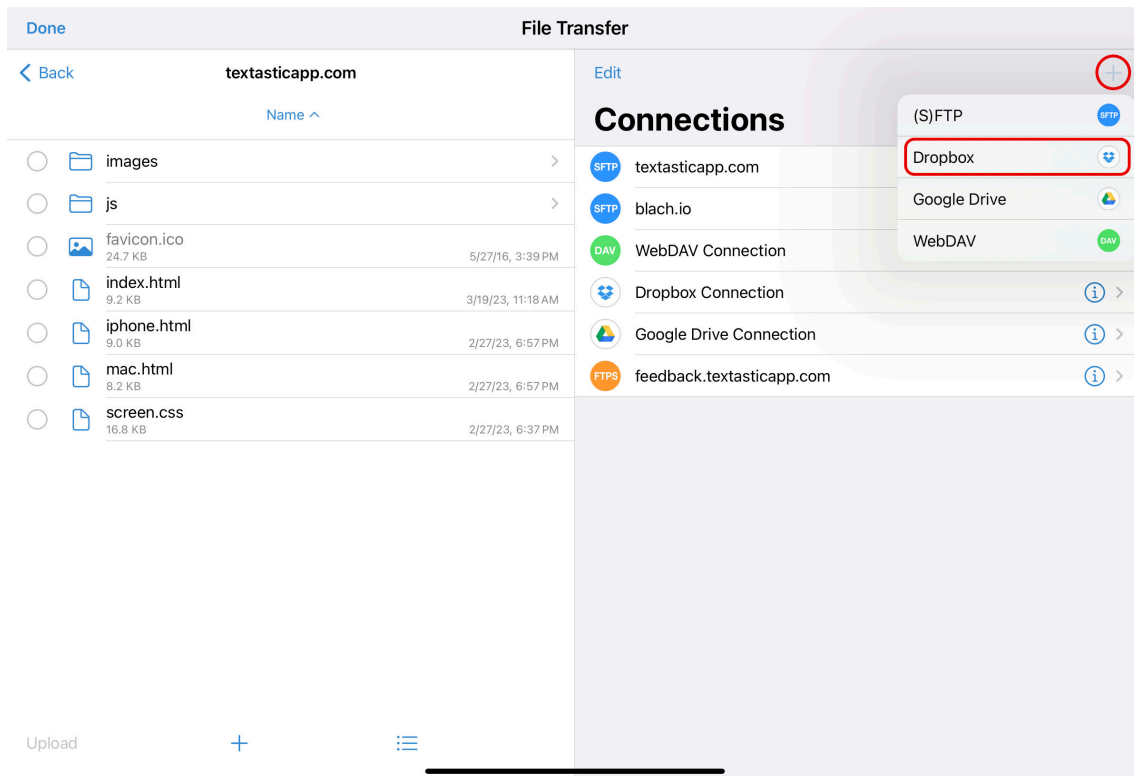
3.3.2 Configure Connection

You can add a custom *Title* that will appear in the connection list (especially useful if you want to connect to several different Dropbox accounts).

3.3.3 Link to your Dropbox account

Tap on the *Account* row to link the remote connection to your Dropbox account.
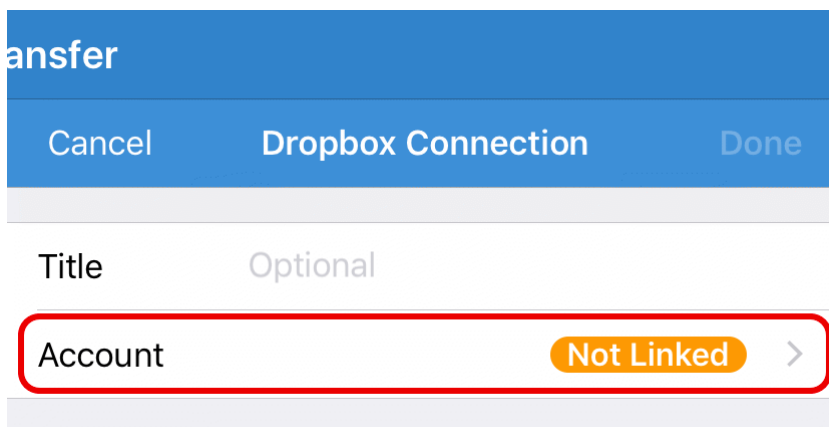
App Authentication

If you have the official Dropbox iOS app installed, this will open the app and ask whether you want to *Allow* Textastic access to your Dropbox.
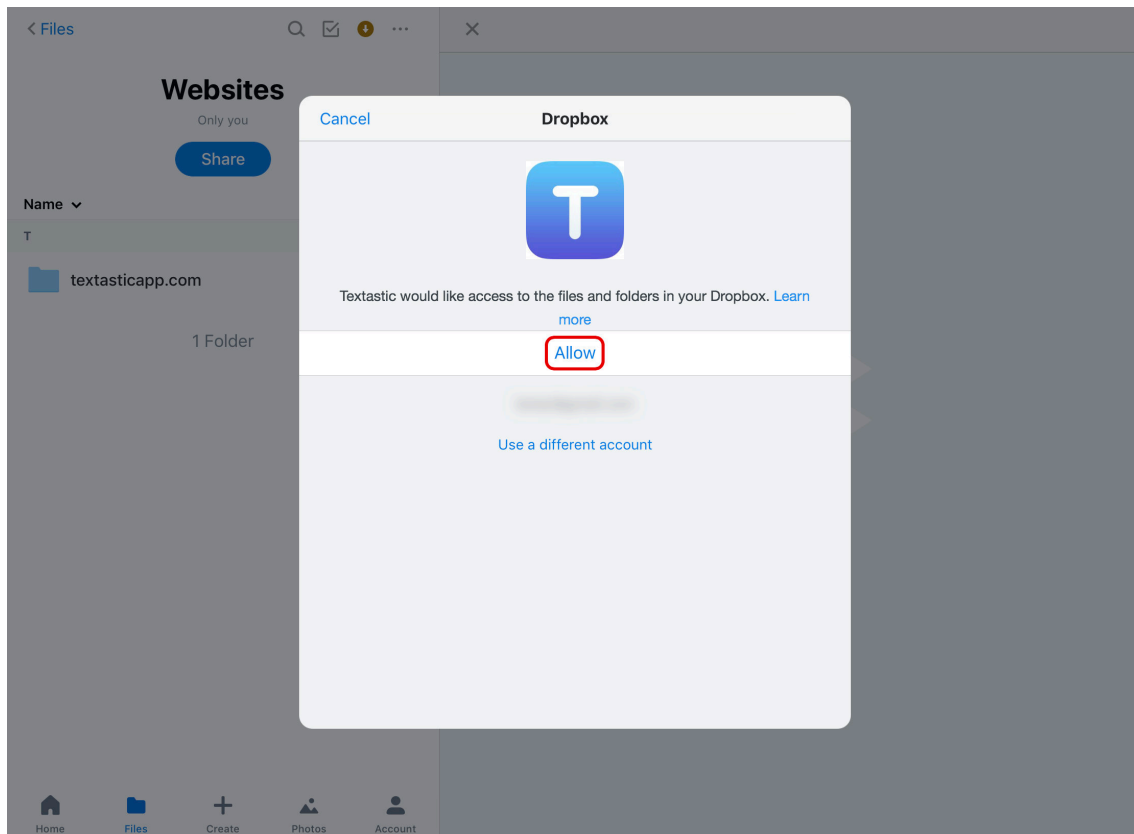
Web Authentication

If instead you do not have the Dropbox installed, a web browser view will open and you will be asked to *Sign in* to your Dropbox account.

Done   〈 〉 🔒 **Dropbox, Inc**

Sign in to Dropbox to link with Textastic

| G | **Sign in with Google** |

—————————— or ——————————

Email

Password

This page is protected by reCAPTCHA, and subject to the Google Privacy Policy and Terms of Service.

Sign in

New to Dropbox? Get the App!

Then you can *Allow* Textastic access to your Dropbox account.
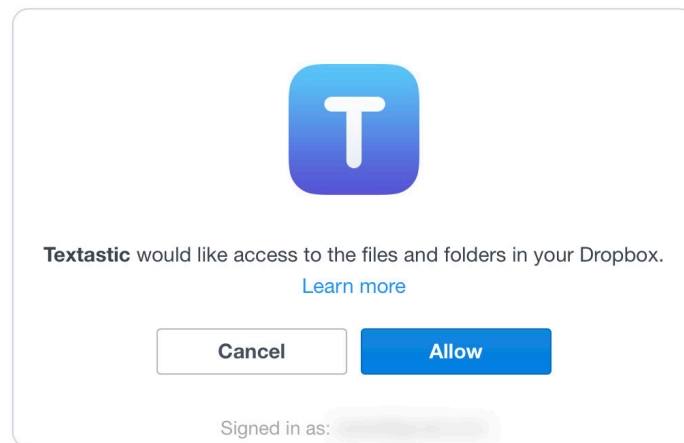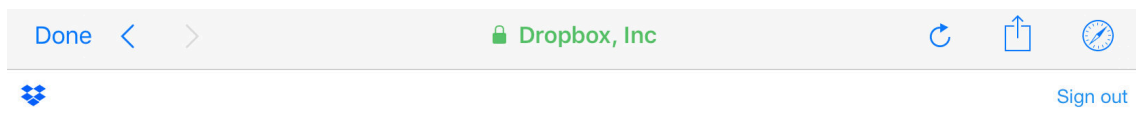
3.3.4 Save Connection

The Dropbox connection is now linked to your Dropbox account.

Tap *Done* to save this connection and return to the list of remote connections.
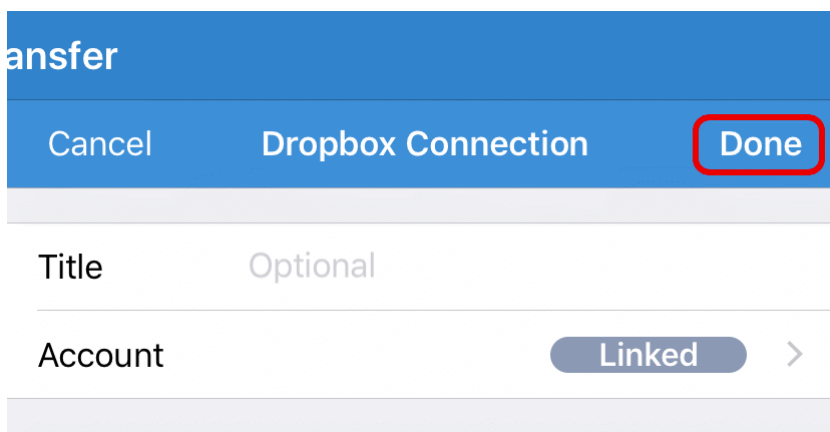
3.3.5 Connect

You can now tap on the newly created connection to connect to your Dropbox account.

3.4 Google Drive Connections

3.4.1 Add a Google Drive Connection

After opening the File Transfer screen, tap on the + button and choose *Google Drive Connection* to add a new connection.
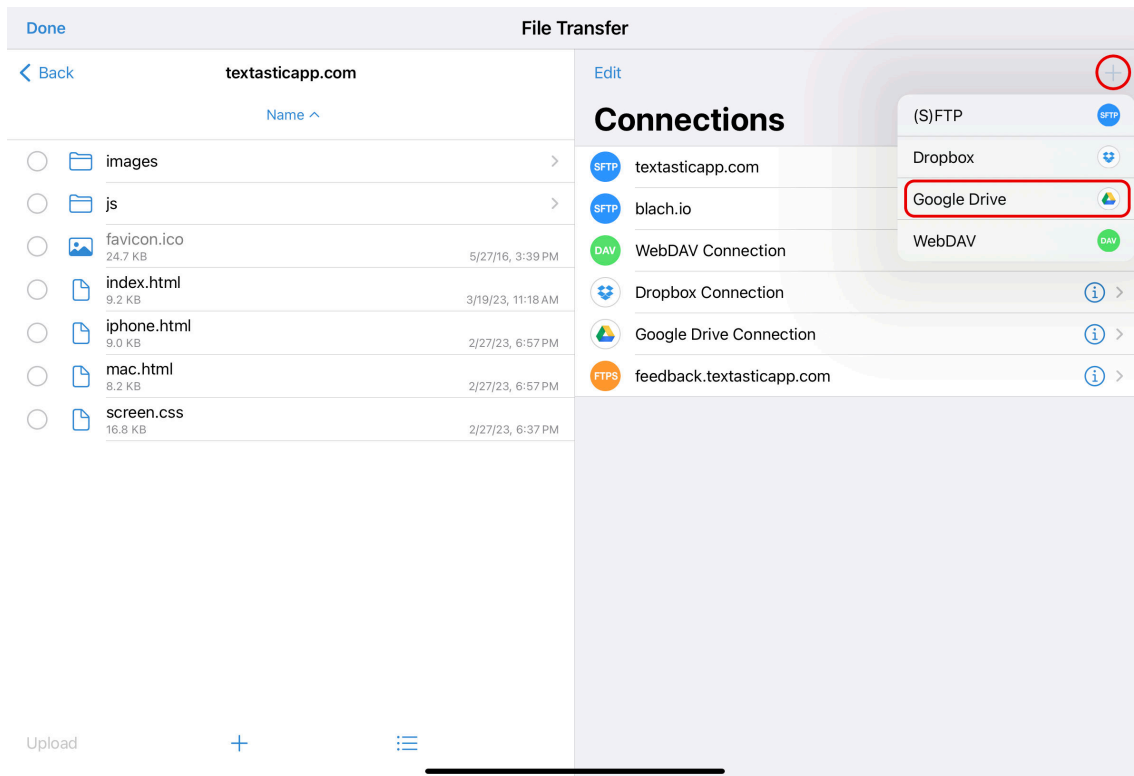
3.4.2 Configure Connection

You can add a custom *Title* that will appear in the connection list (especially useful if you want to connect to several different Google Drive accounts).

3.4.3 Sign in to your Google Drive account

Tap on the *Sign In* row to log in to your Google Drive account.
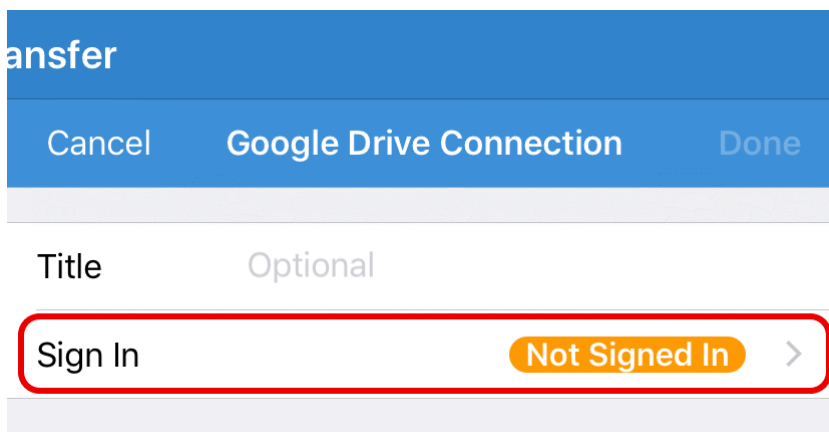
A web browser view will open and you will be asked to *Sign in* to your Google account.

Done ‹ ›   🔒 accounts.google.com   ↻ ⬆ 🧭

G Sign in with Google

Sign in

to continue to **Textastic**

Email or phone

Forgot email?

To continue, Google will share your name, email address, and profile picture with Textastic.

Next

English (United States) ▾   Help   Privacy   Terms

Then you can *Allow* Textastic access to your Google Drive account.
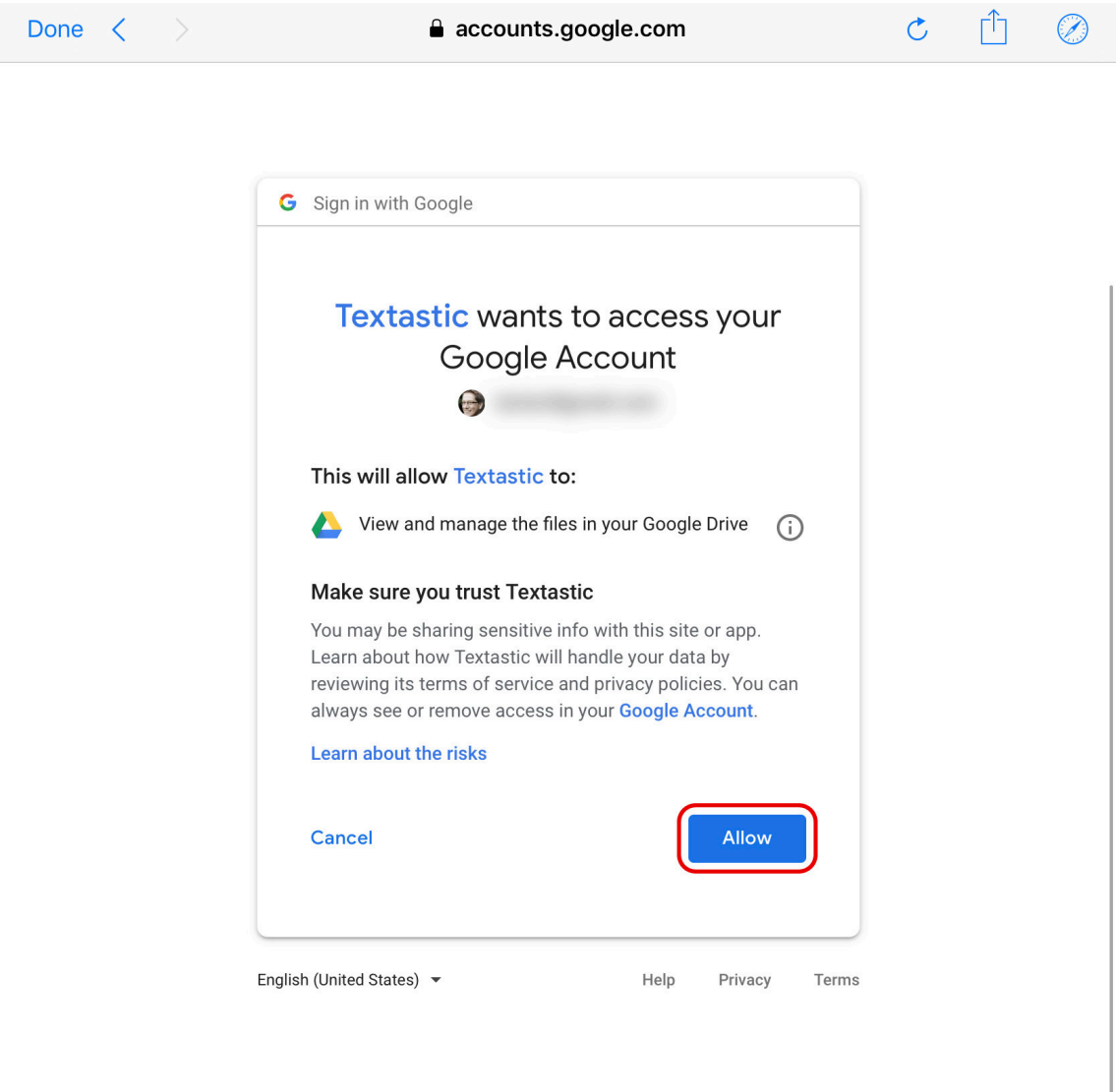
3.4.4 Save Connection

The Google Drive connection is now linked to your Google Drive account.

Tap *Done* to save this connection and return to the list of remote connections.
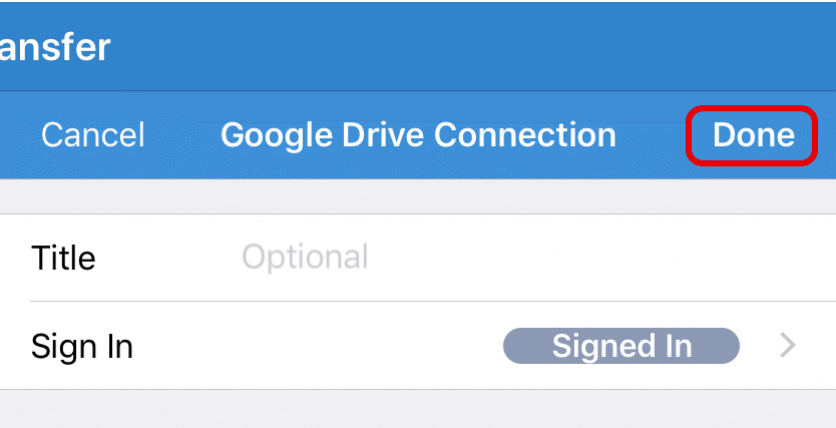
3.4.5 Connect

You can now tap on the newly created connection to connect to your Google Drive account.
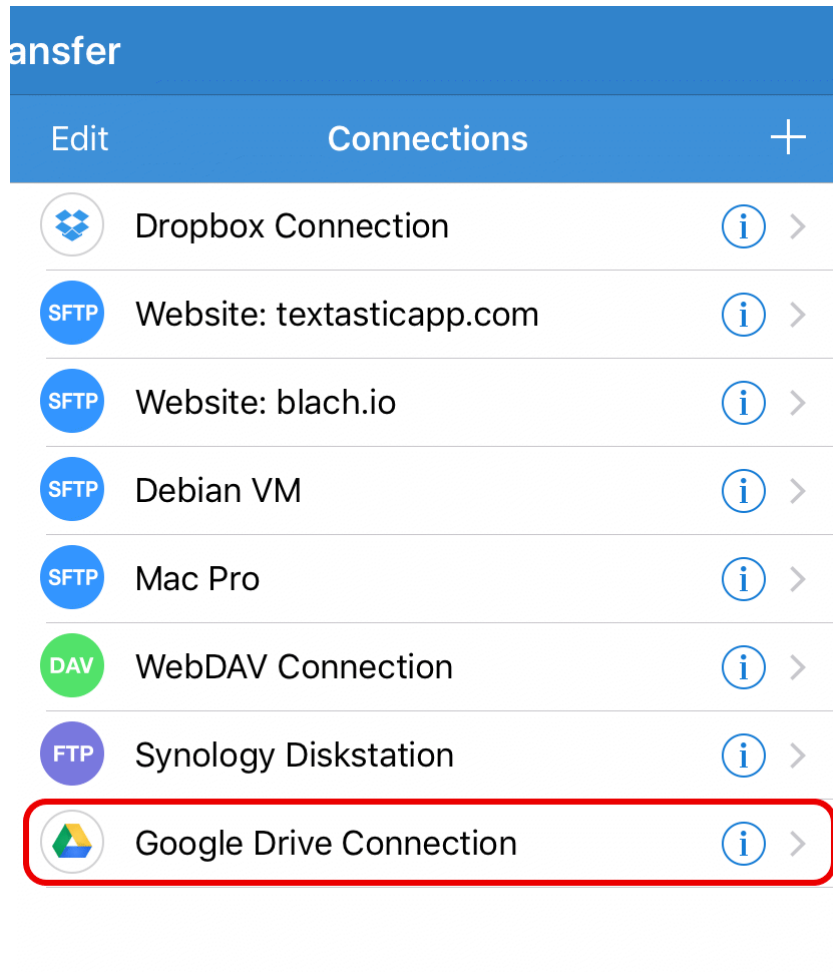
3.5 WebDAV Connections

3.5.1 Add a WebDAV Connection

After opening the File Transfer screen, tap on the "+" button and choose *WebDAV Connection* to add a new connection.
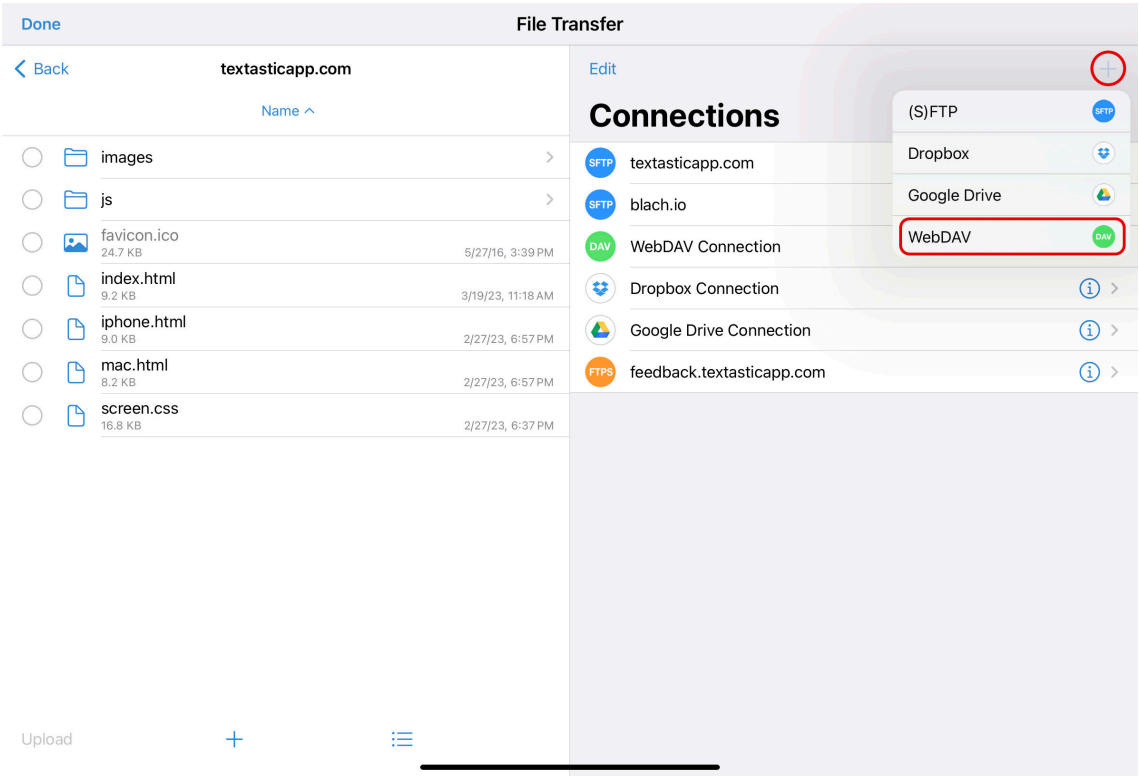
3.5.2 Configure Connection

You can add a custom **title** that will appear in the connection list.

In the **URL** field, enter a valid **http** or **https** WebDAV URL.

If your WebDAV server requires a login, you can enter a **user name** and **password**.

For HTTPS connections you can choose whether Textastic **verifies the SSL certificate** of

the server. Disable this if you use a **self-signed certificate** on the server.

3.5.3 Save Connection

Tap **Done** to save this connection and return to the list of remote connections.

3.5.4 Connect

You can now tap on the newly created connection to connect to your Google Drive account.
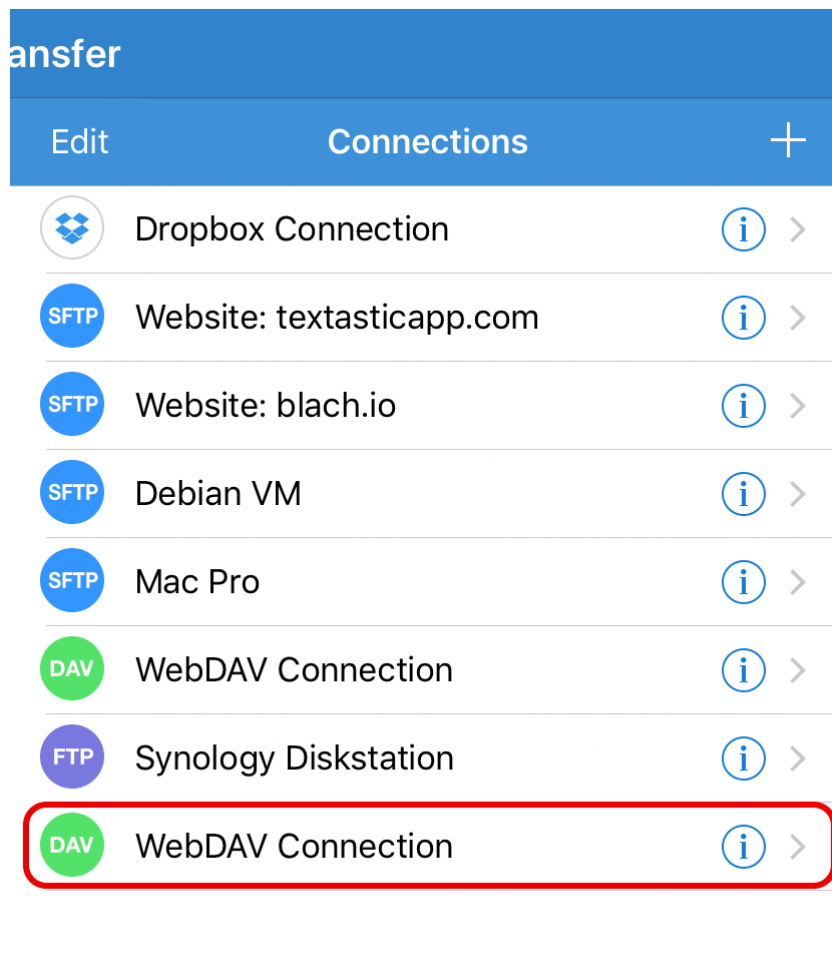
3.6 Uploading and Downloading Files

After you have created a connection to an SFTP, FTPS, FTP or WebDAV server or to Dropbox or Google Drive, you can use it to download and upload files.

First, make sure to open the File Transfer screen.

3.6.1 Select Local Folder

Files you download from a remote server will be downloaded to the current local folder.

Make sure you navigate to the correct local target folder. If no local folder is selected, the *Download* button won't become available.
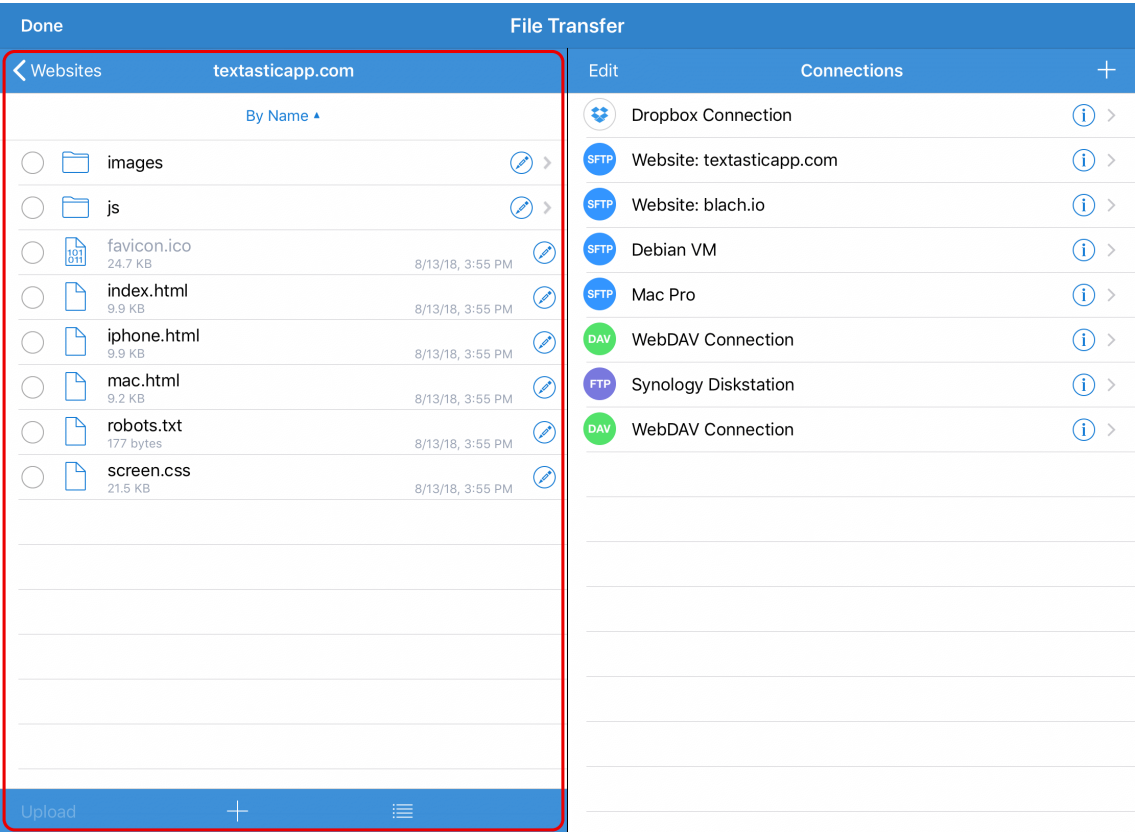
Figure 3.1. Selecting a local folder on iPad
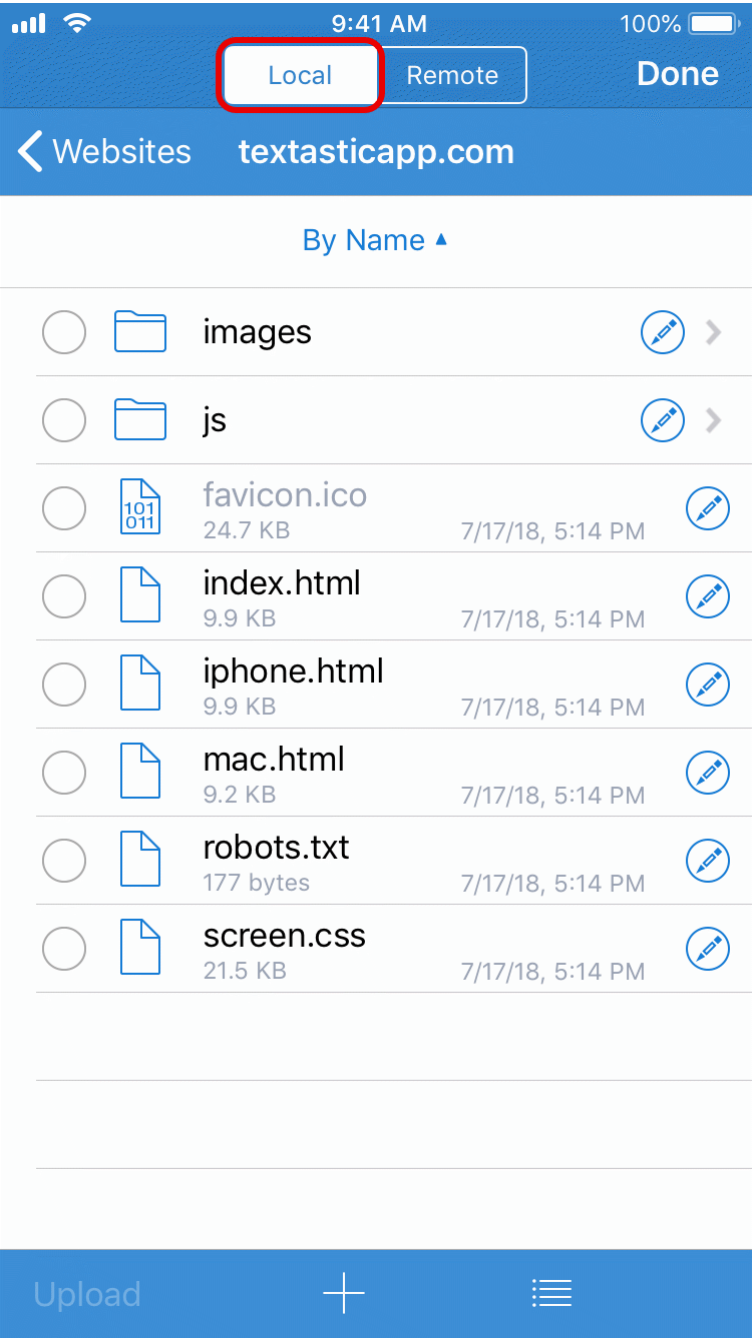
Figure 3.2. Selecting a local folder on iPhone

3.6.2 Connect to Remote Server

Tap on a remote connection to connect to a remote server.
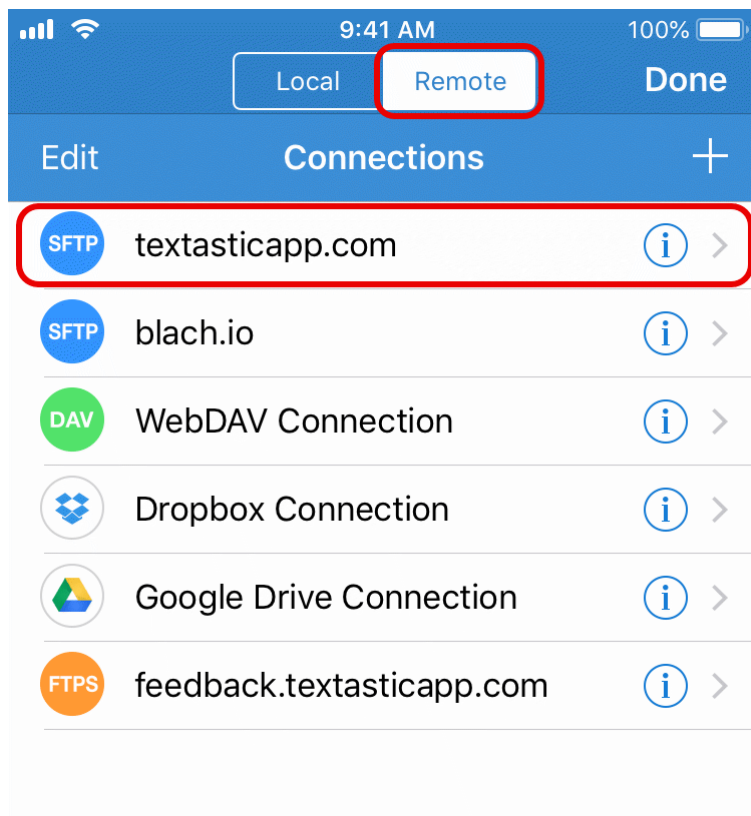
After connecting to the server you should see a list of folders and files. Tap on folders until you reach the location from where you want to download your files.

3.6.3 Downloading Files

Select the files and folders you want to download. Folders are downloaded recursively including all files and subfolders.
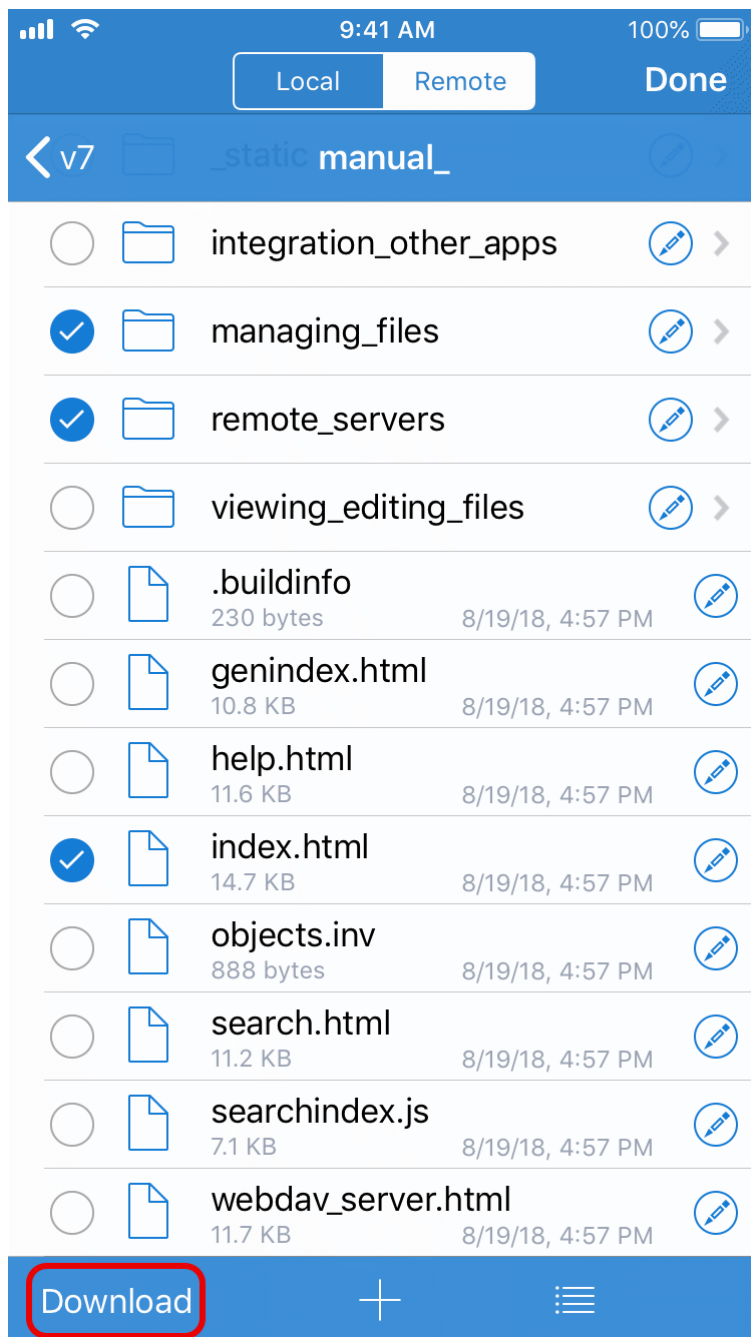
Tap on the *Download* button to start the download. If the button is disabled although you have selected files, make sure to select a local target directory first (see above).

3.6.4 Uploading Files
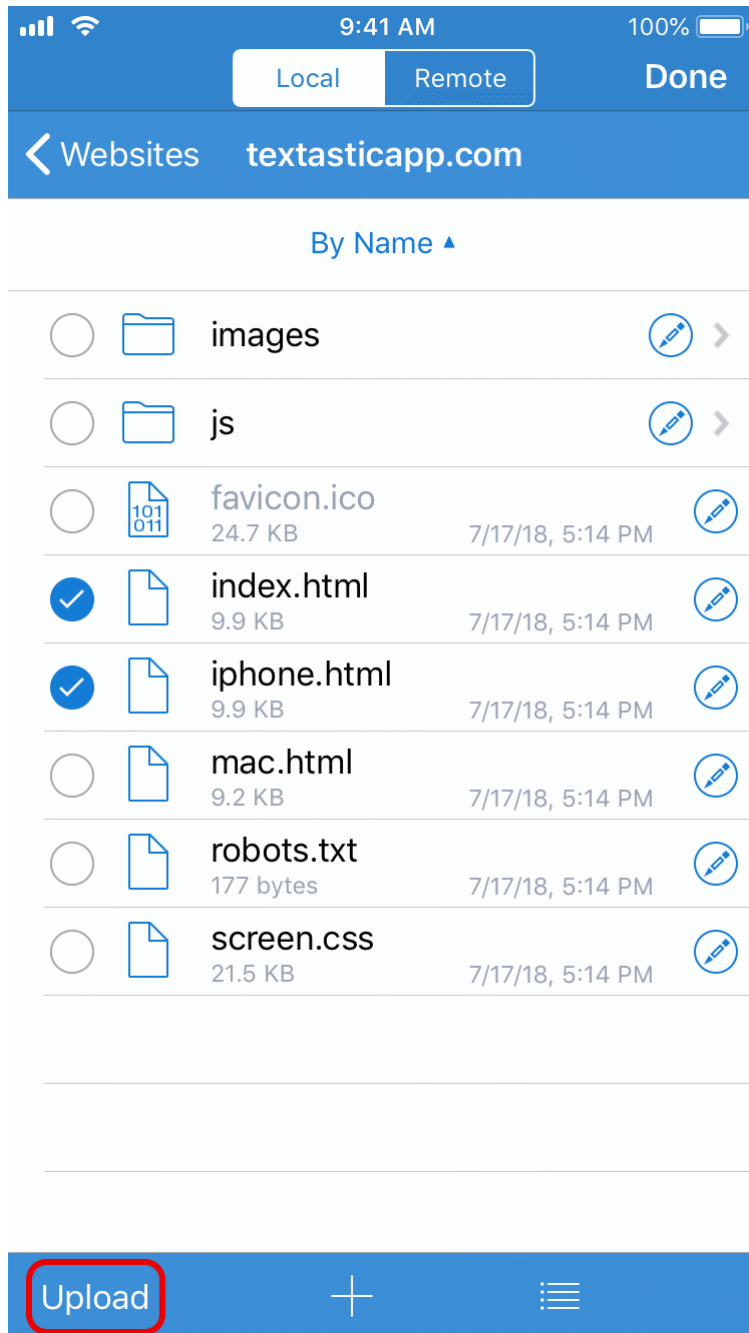
If you want to upload files, select the local files you want to upload and tap on the *Upload* button. If the button is disabled although you have selected files, make sure to connect to a remote server and navigate to a remote target directory first.

3.6.5 Quick uploading and downloading

Once you have downloaded or uploaded files in this way, Textastic stores the remote server and location and you can quickly download the latest version or upload your changes. See: Quickly Uploading and Downloading Files.

3.7 Quickly Uploading and Downloading Files

Textastic offers a fast and easy way to upload files you already downloaded from a remote server. You can also download the latest version if you know that the file was changed on the remote server.

3.7.1 Show Linked Remote Connections

When you upload files to or download files from a remote server (SFTP, FTPS, FTP, WebDAV, Dropbox, or Google Drive), Textastic saves the connection used for the download as well as the remote path of the file. You can see this information in the *File Properties*.

You can save several remote locations for each file by going to the File Transfer screen and manually uploading the file to another location, or downloading the file from another location and overwriting the local file.

Tap on the **path** to see the full remote path.

3.7.2 Quick Upload or Download

When a file is linked to a remote connection, there are two new options when you tap on the *More* button in the top right corner:
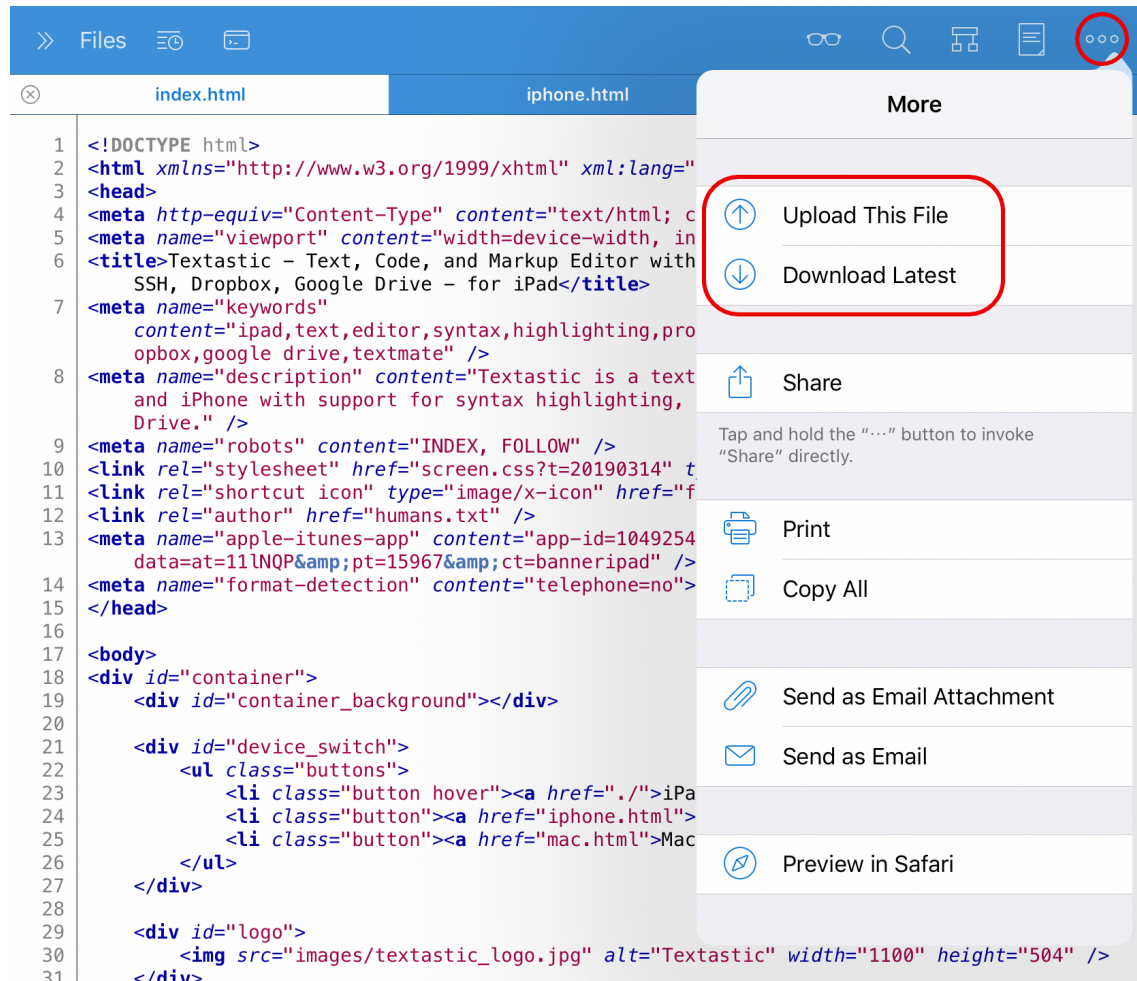
Upload This File: This uploads the currently open file to the same remote location it was originally downloaded from.

Download Latest: This will connect to the remote server and try to download the latest version of the file.

If there are links to more than one remote connection, Textastic will ask you which one to use:
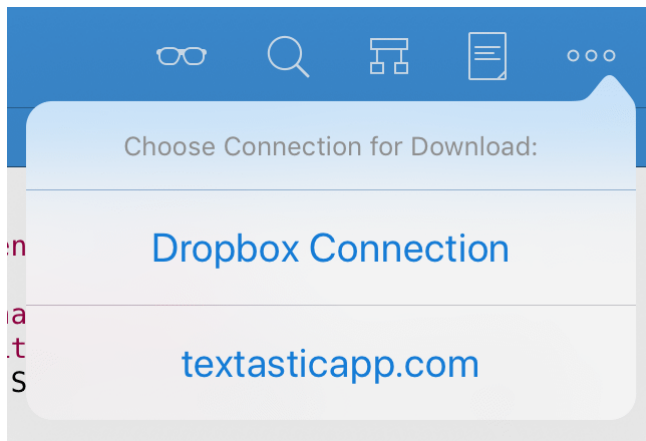
3.8 SSH Terminal Connections

Textastic has a built-in SSH client and terminal emulator. You can use it to open SSH terminal connections in tabs. Multiple simultaneous SSH session are supported.

3.8.1 Selecting an SSH Connection

Tap on the *Terminal* button to show a list of *SSH connections*.
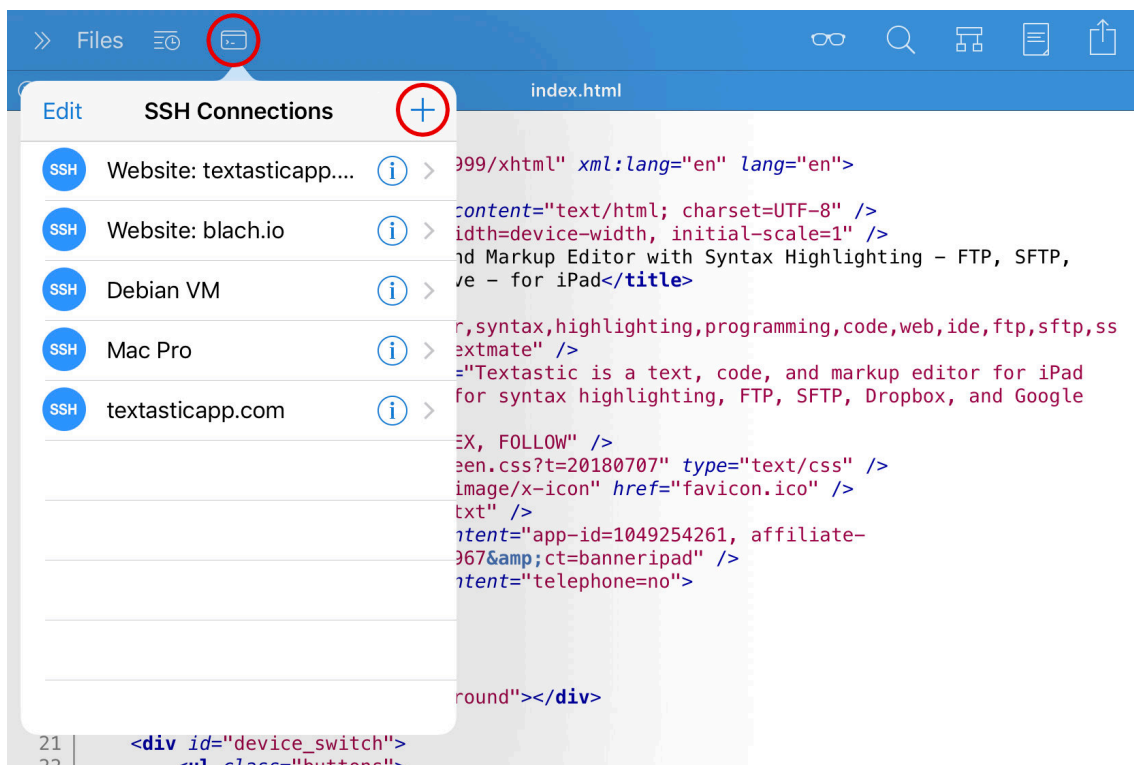
If you have already configured Section 3.2.4 connections, they can be re-used for an SSH terminal connection.

If there is no SSH/SFTP connection yet, tap on the + button to add a new one.
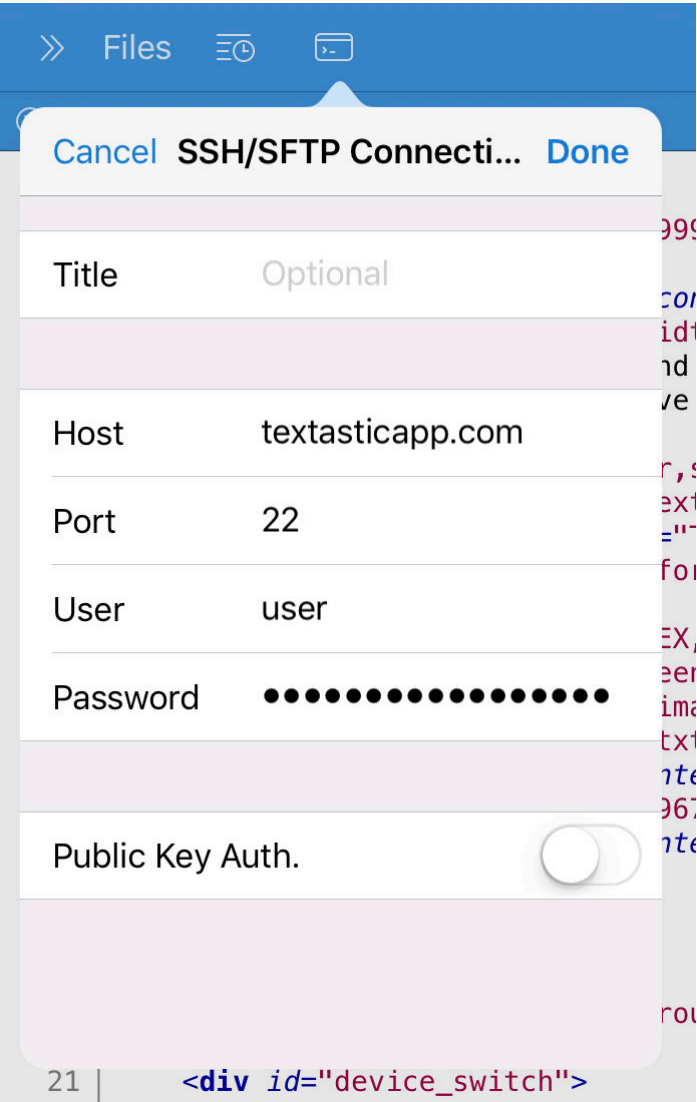
Please have a look at Section 3.2.4 for details on how to configure the connection.

3.8.2 Connecting to an SSH server

Tap on an SSH connection to open a new terminal tab and connect to the SSH server.
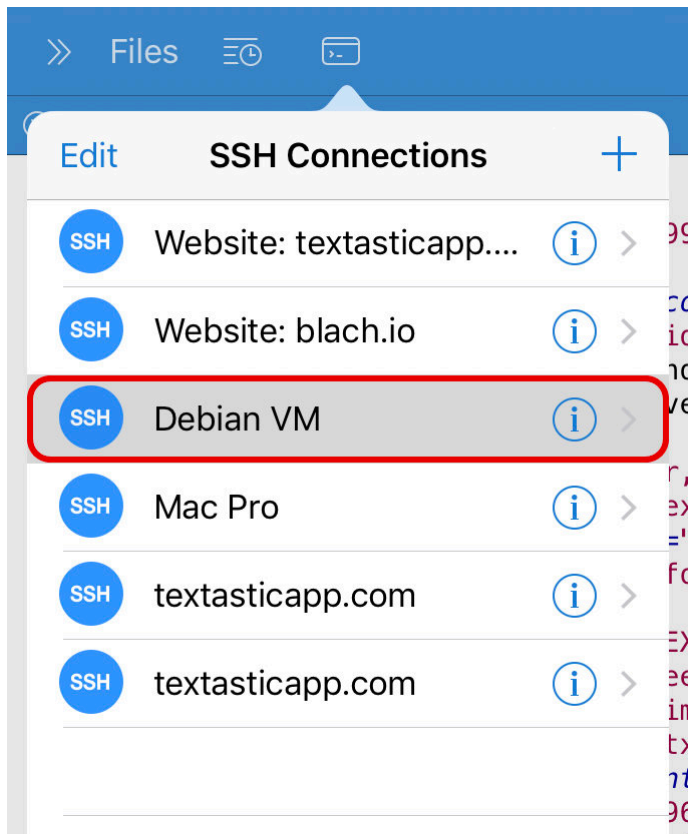
Once connected, you can use the keyboard to enter commands.
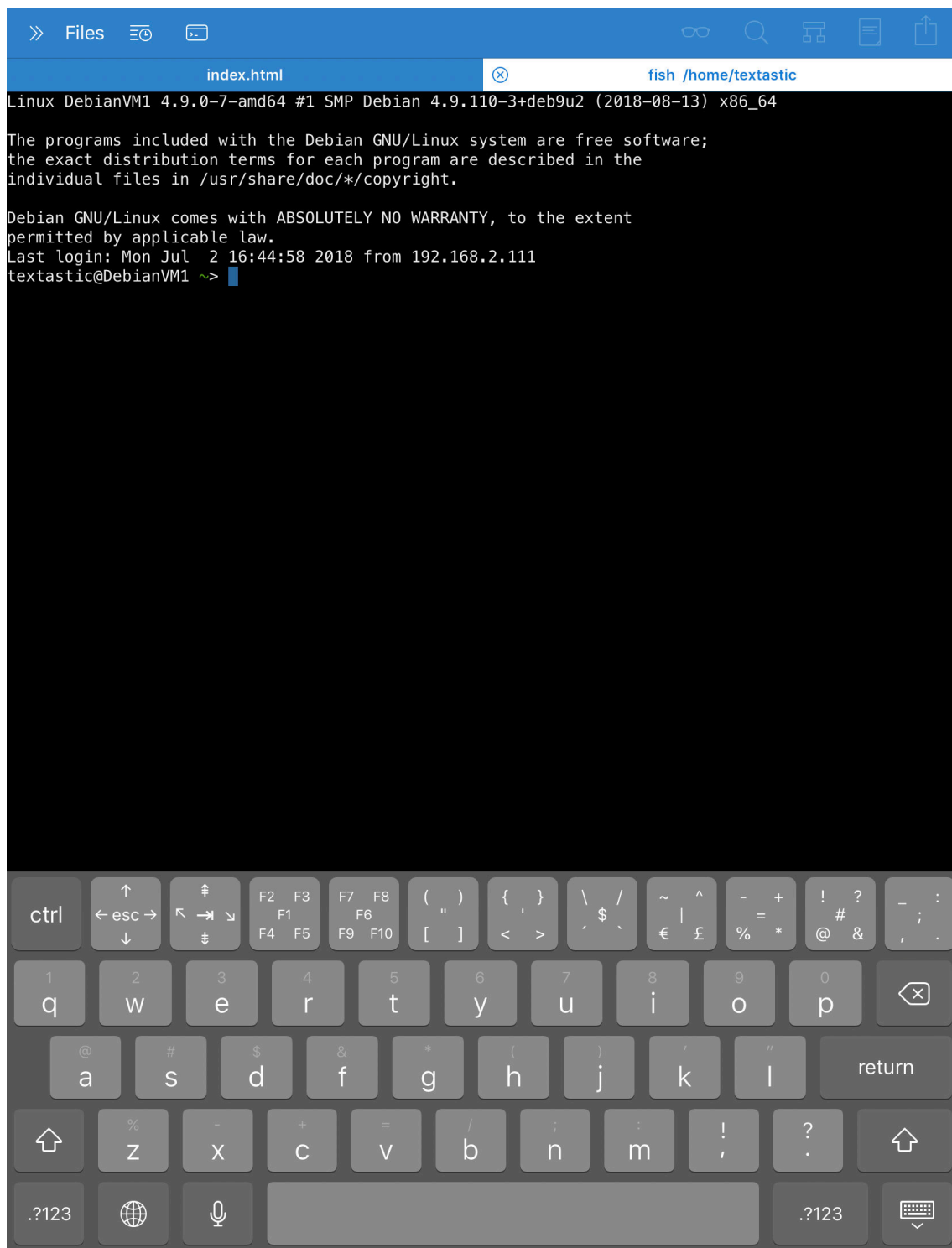
3.8.3 Special On-Screen Keyboard Keys

When a terminal tab is active, there is a special row of keys above the standard on-screen keyboard.

Figure 3.3. SSH keyboard row on iPad

Figure 3.4. SSH keyboard row on iPhone

If you want to enter the character in the middle of the button, just tap on the button. If you want to enter one of the other four characters, make sure to **swipe over the button in the direction of the character** you want to type.

Control

Tap on the *ctrl* key and then on one of the regular characters of the keyboard to send a control character. For example, you can press *ctrl* and *d* to send the EOT (end of transmission) character and end the SSH session.

Escape and cursor keys

Tap on the *esc* key to send the Escape character. In most cases, the Esc key also does the same thing as the **Alt key**.

Swipe in the direction of the arrows to send cursor left, right, up, and down commands.

Tab, Page Up/Down, Home, End

Tap on the ⇥ key to send the Tab character. Swipe left and right for Home and End. Swipe up and down for Page Up and Page Down.

F1 to F10

Tap on the *F1* and *F6* button or swipe in the direction of the other F-Keys to send a Function key command.

Hide keyboard

The right-most button can be used to hide the keyboard. *(only on iPhone)*

The rest of the buttons can be used to quickly send the special characters displayed on the button.

3.8.4 Shortcuts on an External Keyboard

The following special keys on a hardware keyboard are supported:

- `Tab`, `Escape`, and `Arrow` keys

- `Control` modifier key

- use `Cmd` and the arrow keys to send Home (`Cmd-left`), End (`Cmd-right`), Page Up/Down (`Cmd-up`, `Cmd-down`)

- use `Cmd-1` to `Cmd-0` to enter F1 to F10

4 Integration with Other Apps

4.1 External Files and Folders

In addition to the built-in clients for (S)FTP, WebDAV, Dropbox, and Google Drive, you can also use the **iOS document picker** to open files from a huge number of third party file providers.

4.1.1 Open Files and Folders Using the iOS Document Picker
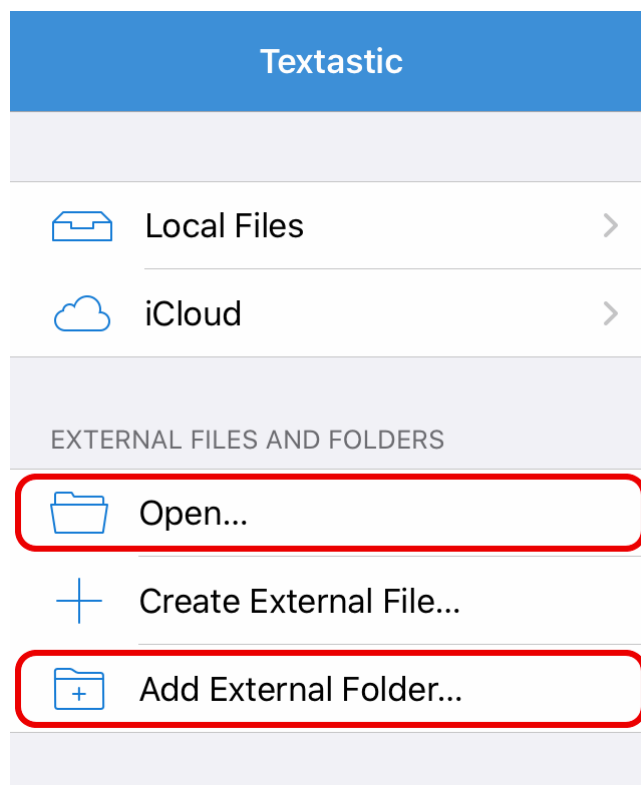
Opening Files

Tap *Open…* or press `Cmd-O` on an external keyboard to open files using iOS document picker.

Select a location on the left and **tap on a file** on the right to open it in Textastic.
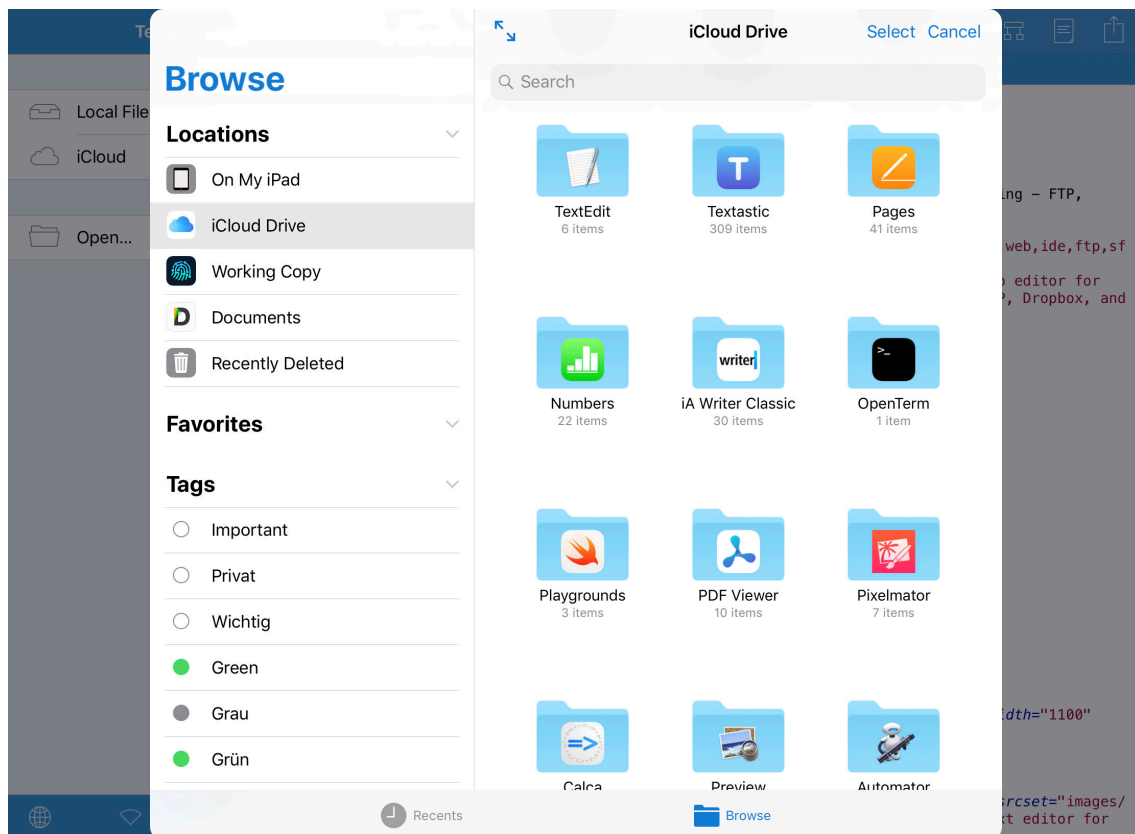
You can also tap on the *Select* button in the top-right corner and select **multiple files** to open them in tabs.

Files opened in this way are not imported. Instead, they are **opened in-place**. Changes you make to the files are automatically saved to their original location.

Note If you plan to use the built-in web preview to preview a HTML file with **external JavaScript, CSS, or image files**, it is recommended to use one of the built-in clients for (S)FTP, WebDAV, Dropbox, or Google Drive to download your website files first and edit and preview them locally. If you open a single file with the document picker, linked files cannot be accessed by the web preview due to the iOS security sandbox that limits an app's access to files in other apps.

Adding External Folders

For some file providers it's also possible to open folders. This currently works with the Git client **Working Copy**, the SFTP client **Secure ShellFish**, folders in **iCloud Drive**, and the **"On my iPad/iPhone"** location.

Note Other file provider apps (like **Google Drive** or **Dropbox**) would need to be updated by their developers to support this. Those locations are currently disabled in the folder picker by iOS. You can however use the *Open…* command (see above) to open individual files from these file providers.

As an alternative you can use Textastic's built-in Google Drive and Dropbox clients to download whole folder hierarchies.

- Tap *Add External Folder…* to add a folder using the iOS folder picker.
- Select a folder.

- Tap *Done*.
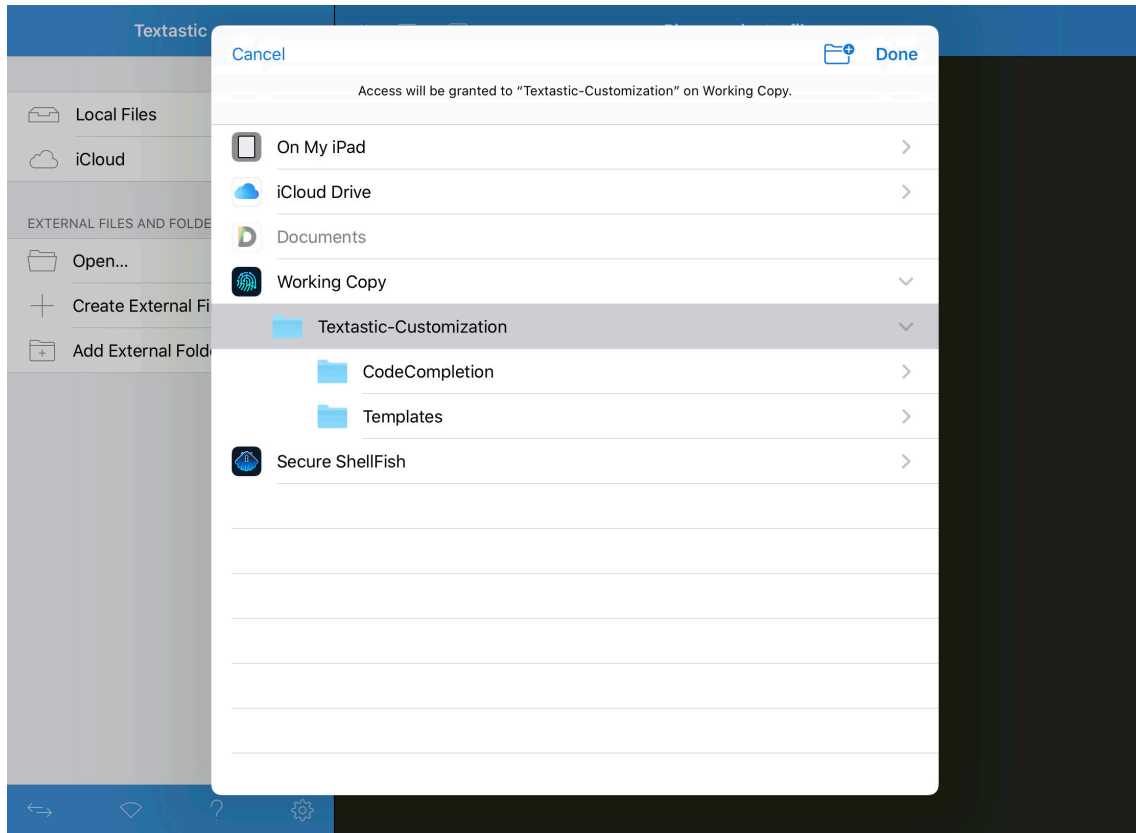
The selected folders will be opened as external folders and behave the same as the Local Files and iCloud locations.
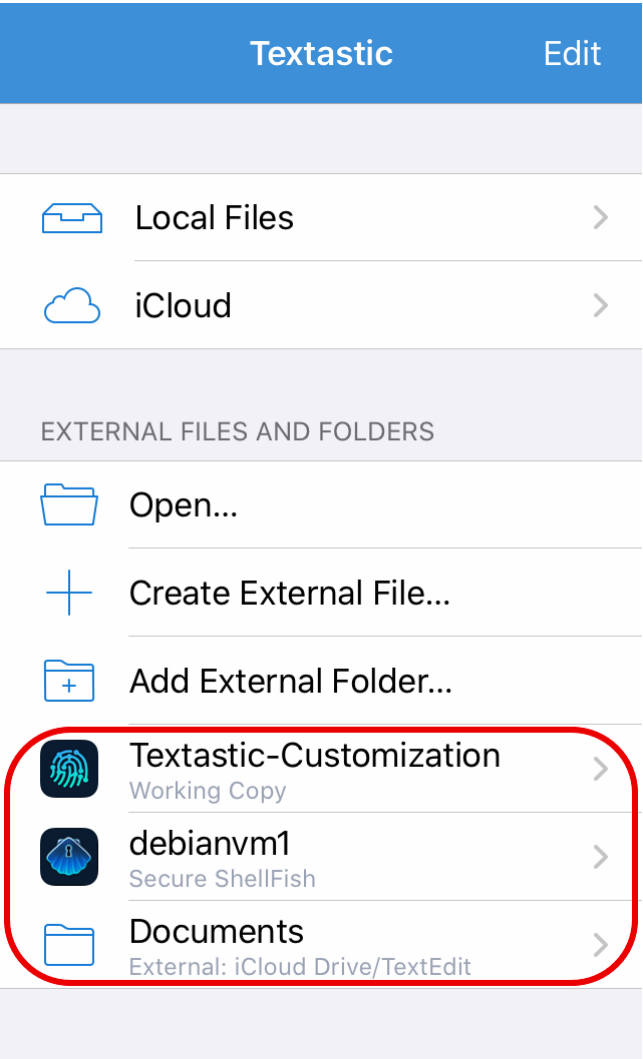
Renaming and Removing External Folders

Invoke the context menu by taping and holding or right-clicking an external folder and choose *Rename*:
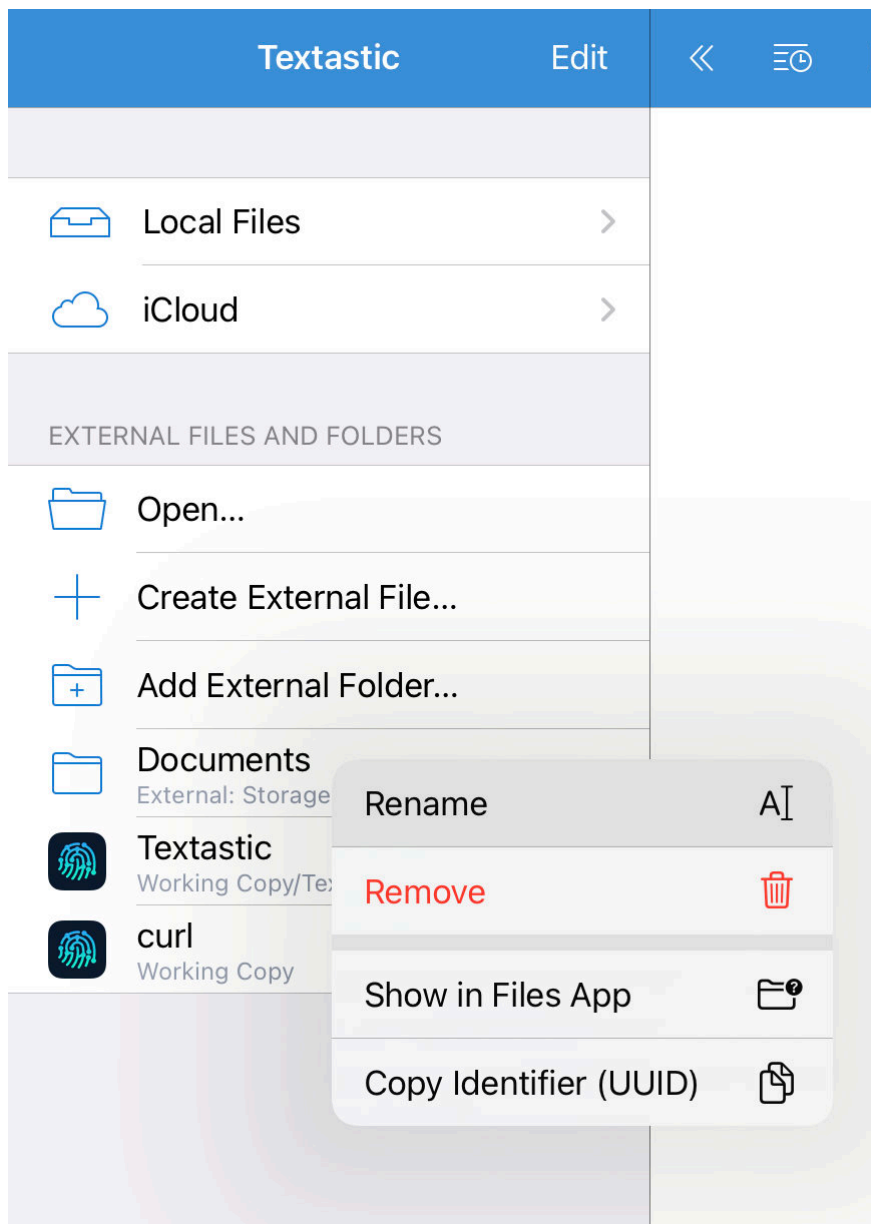
You can also *Remove* an external folder from the context menu or use the *Edit* button or a swipe gesture to remove external folders.

Add Third-Party File Providers

By default, the Files app only allows you to open files from iCloud Drive and the "On my iPad/iPhone" location. However, there are many third-party apps with file providers that integrate with the Files app:

- Microsoft OneDrive
- Dropbox
- Box
- Adobe Creative Cloud
- Google Drive
- Working Copy (Git client)

- Seafile

- OwnCloud

- *… and many more*

Follow these steps on your iPhone, iPad, or iPod touch with iOS 11 or later:

- Download and set up the third-party cloud app.

- Open the **Files** app.

- Tap *Locations > Edit*.

- Slide to turn on the third-party apps that you want to use in the Files app.

- Tap *Done*.

If you don't see one of your third-party cloud services in the Files app, check with your cloud provider.

Please also have a look at this Apple support article to see how to use the **Files** app on iOS.

4.2 Git Client Working Copy

Textastic works great with the Git client Working Copy. Git Repositories you clone in Working Copy can be opened as **external folders** in Textastic.

External folders behave just like regular local folders in Textastic and can even be used in the File Transfer screen to upload and download files using the built-in remote connections.

Changes you make to external folders are automatically reflected in the source app. So for example, if you edit a file in a repository opened as an external folder, the file status will automatically switch to "modified" in Working Copy.

4.2.1 Use the Folder Picker to Open a Repository

Use the *Add External Folder…* command to add a Git repository from **Working Copy** as an external folder to Textastic as described at Adding External Folders.

4.2.2 Use Drag and Drop on iPad to Open a Repository

On iPad, you can also drag a repository from Working Copy into the locations screen of Textastic to add it as an external folder:
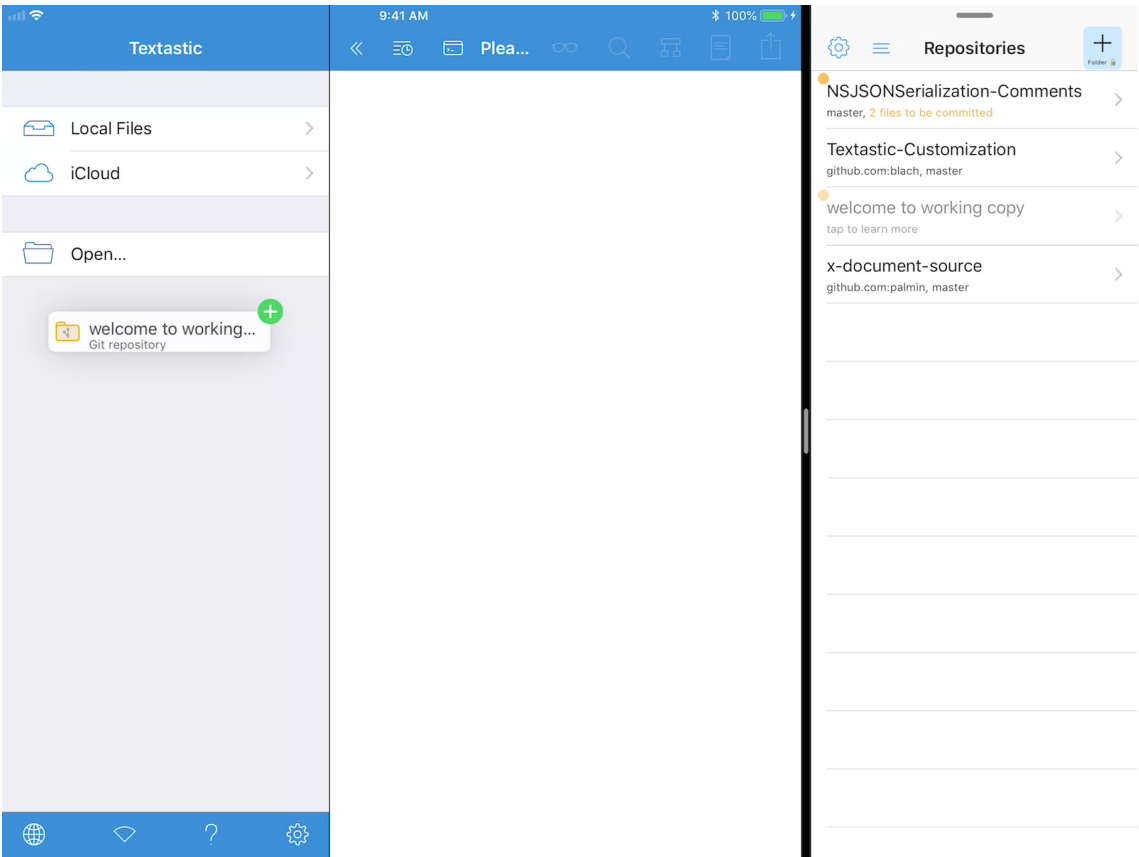
You can either do this in **Split View** as shown above or by starting a drag in Working Copy, using the **Dock** or **app switcher** to switch to Textastic, and drop it into the locations screen.

Important For this to work, `Working Copy` must be added as a location in the **Files** app.

You can read more about Multitasking on iPad, Split View and Drag and Drop in this Apple support article.

4.3 X-Callback-URL API and Other URL Schemes

Textastic for iOS exposes actions that can be used by other apps using the x-callback-url specification.

All currently implemented actions involve opening an existing file or creating a new one and inserting or replacing text.

4.3.1 Registered URL schemes

To target any installed version of Textastic, use the `textastic://` URL scheme.

These additional URL schemes are registered by the various versions of Textastic:

App	URL Scheme
Textastic v6 and later (universal app)	x-textastic-universal://
Textastic v5 and earlier for iPad	x-textastic-ipad://
Textastic v5 and earlier for iPhone	x-textastic-iphone://

The "x-" versions are provided if you need to specifically call or test for the installation of the Universal vs. the iPad or iPhone version.

4.3.2 *x-callback-url* URL Format

The format for Textastic's URL scheme looks like this:

```
textastic://x-callback-url/[action]?[x-callback     parameters]&[action
parameters]
```

The **x-callback parameters** are defined by the *x-callback-url* protocol, and are optional for all actions.

The **action parameters** are defined in the API below, and are unique to each action.

Some actions will return information to the calling app. When an action returns information, the parameters passed back to the calling app are documented with the API below.

4.3.3 Supported *x-callback-url* Parameters

All of Textastic's actions will accept all of the x-callback parameters: `x-source`, `x-success`, `x-error`, and `x-cancel`. There are some cases where the action and callback do not make sense. For example, when the open action successfully opens a file, there is no reason to callback the x-success URL. In these cases, the x-callback parameter is ignored.

- **x-source**: The name of the calling app.
- **x-success**: A complete URL to call when the action completed successfully.
- **x-error**: A complete URL to call when the action failed.
- **x-cancel**: A complete URL to call when the action was canceled by the user.

4.3.4 Actions

new

Create a new file in the local file system or in iCloud.

If a file with the specified name already exists, Textastic will generate a unique file name by appending a number to the file name (e.g. "test 2.txt" instead of "test.txt").
Parameters

- **location**: *iCloud*, *local*, or *external*. If not specified, defaults to the local file system.
- **path**: The directory path. If not specified, defaults to the root folder.
- **name**: The name of the file to create. If a file with that name already exists, Textastic will append a number to the file name to make it unique. If not specified, a file name will be generated by Textastic.
- **externalUUID**: If location is "external", use this parameter to specify the identifier of an external folder. You can get the UUID by showing the context menu of an external folder in Textastic and selecting "Copy Identifier (UUID)".
- **text**: The text to insert into the new file.
- **snippet**: The snippet to insert. If both the *snippet* and the *text* parameter are present, only the snippet parameter will be used.

If neither the *text* parameter nor the *snippet* parameter is specified, the text to append will come from the clipboard.

Examples

Create a new file "foo.txt" in the local root directory and insert "bar":

```
textastic://x-callback-url/new?name=foo.txt&text=bar
```

Create a new file "foo.txt" in iCloud in the "test" folder and insert the snippet "foo $0 bar". This snippet places the cursor between foo and bar after the insertion.

```
textastic://x-callback-url/new?location=icloud&name=foo.txt&path=test&snippet=foo%20$
```

Create a new file "test.txt" in the specified external folder in the "test" folder and insert the clipboard contents.

```
textastic://x-callback-url/new?location=external&externalUUID=62E8D362-7DFA-413C-874E
```

---

open

Open an existing file in the local file system, iCloud, or an external folder.

If the file doesn't exist, calls the url from the x-error parameter.
Parameters

- **location**: *iCloud*, *local*, *external*, or *fullPath*.
  If not specified, defaults to the local file system.
  Use "fullPath" to open an external file if you don't know the UUID of an external folder, but have the full local path of the file in the file system. This is useful for apps that include a file provider and want to open a file in Textastic. It is used by the "Edit in Textastic" command of the Git client app Working Copy.
- **path**: The directory path. If not specified, defaults to the root folder.
- **name**: The name of the file to open.
- **externalUUID**: If location is "external", use this parameter to specify the identifier of an external folder. You can get the UUID by showing the context menu of an external folder in Textastic and selecting "Copy Identifier (UUID)".
- **suggestedExternalFolderPath**: If location is "fullPath", you can specify this parameter and Textastic will prompt for adding that folder as an external folder if the file could not be accessed.

Example

Open the file "index.html" in the "example.com" folder in the local file system.

```
textastic://x-callback-url/open?path=example.com&name=index.html
```

---

append

Open an existing file or create a new file and append text.
Parameters

- **location**: *iCloud*, *local*, or *external*. If not specified, defaults to the local file system.
- **path**: The directory path. If not specified, defaults to the root folder.
- **name**: The name of the file to open. If the file doesn't exist, it will be created.
- **externalUUID**: If location is "external", use this parameter to specify the identifier of an external folder. You can get the UUID by showing the context menu of an external folder in Textastic and selecting "Copy Identifier (UUID)".

- **text**: The text to append to the file.
- **snippet**: The snippet to append to the file. If both the *snippet* and the *text* parameter are present, only the snippet parameter will be used.

If neither the *text* parameter nor the *snippet* parameter is specified, the text to append will come from the clipboard.
Example

Append the contents of the clipboard to the file clipboard.txt in the iCloud root folder:

```
textastic://x-callback-url/append?location=iCloud&name=clipboard.txt
```

---

replace

Open an existing file or create a new file and replace its contents with the specified text.
Parameters

- **location**: *iCloud*, *local*, or *external*. If not specified, defaults to the local file system.
- **path**: The directory path. If not specified, defaults to the root folder.
- **name**: The name of the file to open. If the file doesn't exist, it will be created.
- **externalUUID**: If location is "external", use this parameter to specify the identifier of an external folder. You can get the UUID by showing the context menu of an external folder in Textastic and selecting "Copy Identifier (UUID)".
- **text**: The text to insert into the file after its contents are deleted.
- **snippet**: The snippet to insert into the file after its contents are deleted. If both the *snippet* and the *text* parameter are present, only the snippet parameter will be used.

If neither the *text* parameter nor the *snippet* parameter is specified, the replacement text will come from the clipboard.
Example

Replace the contents of the file scratchpad.txt in the iCloud root folder with the string "foo":

```
textastic://x-callback-url/replace?location=iCloud&name=scratchpad.txt&text=foo
```

---

reloadCustomizations

Reload customizations (custom syntax definitions, themes, and code completions) from the "Local Files/#Textastic" folder.
Parameters

None.
Example

```
textastic://x-callback-url/reloadCustomizations
```

4.3.5 Download using the textastic:// scheme

In addition to *x-callback-url* support, the `textastic://` scheme can be used to easily download the server response of HTTP URLs.

You can use this feature to **view the source code of a website** or download files into Textastic.

This is what you need to do:

- Make sure your iPad or iPhone has an active internet connection
- Open a website in **Safari** or another browser app like **Chrome**
- Tap on the address bar
- **Replace** `http://` **with** `textastic://`
- Tap *Go* on the on-screen keyboard
- Textastic will open and start to download the file at the URL. It will be saved to the root directory in the local file system. When the file is downloaded, it will be opened in the editor.

4.4 Shortcuts

Textastic adds two actions to Apple's Shortcuts app: "Get File" and "Open File".
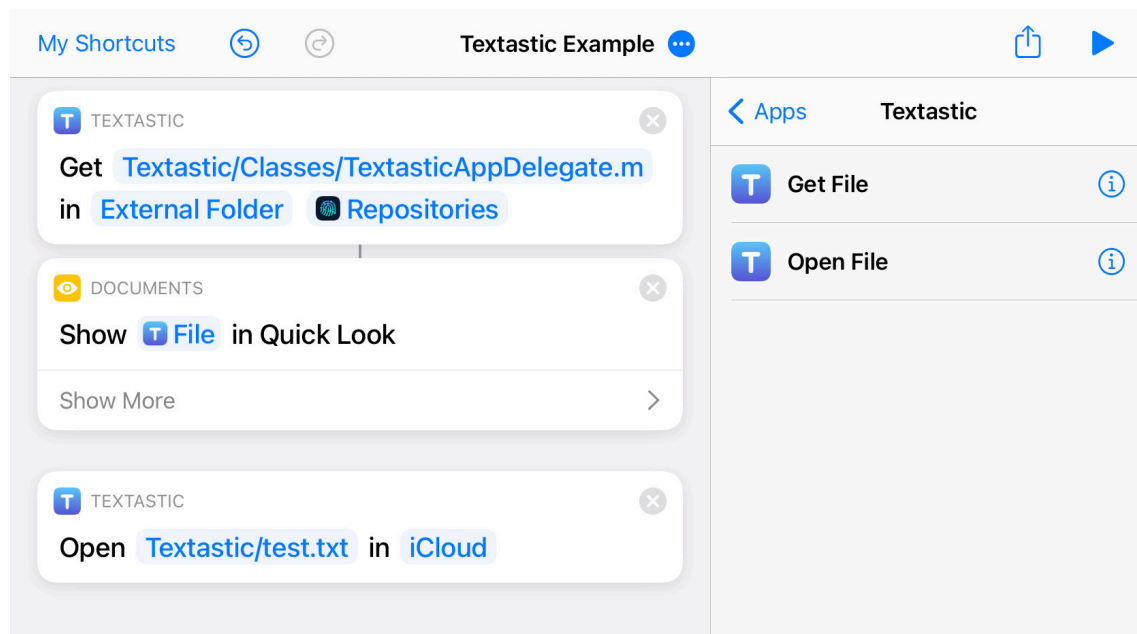
4.4.1 Get File

With this action, you can get the contents of any file available in Textastic. You can enter the path of a file in Local Files, iCloud and external folders.
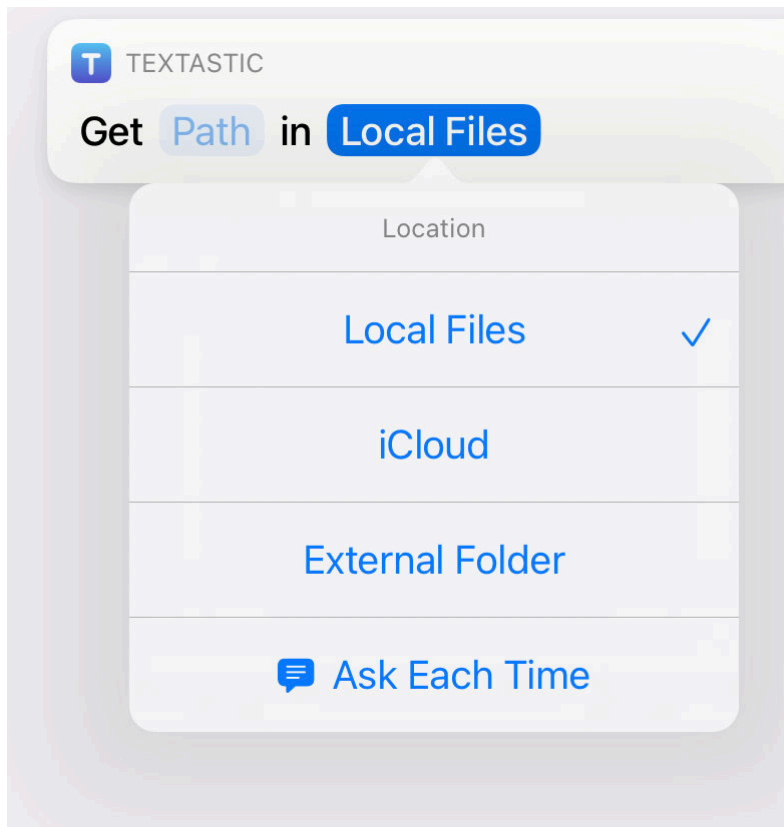
4.4.2 Open File

Use this action to open an existing file in Textastic.

4.4.3 Path Parameter

To easily get the correct value for the path parameter, you can open the context menu of a file in Textastic and select **"Copy Path"**:
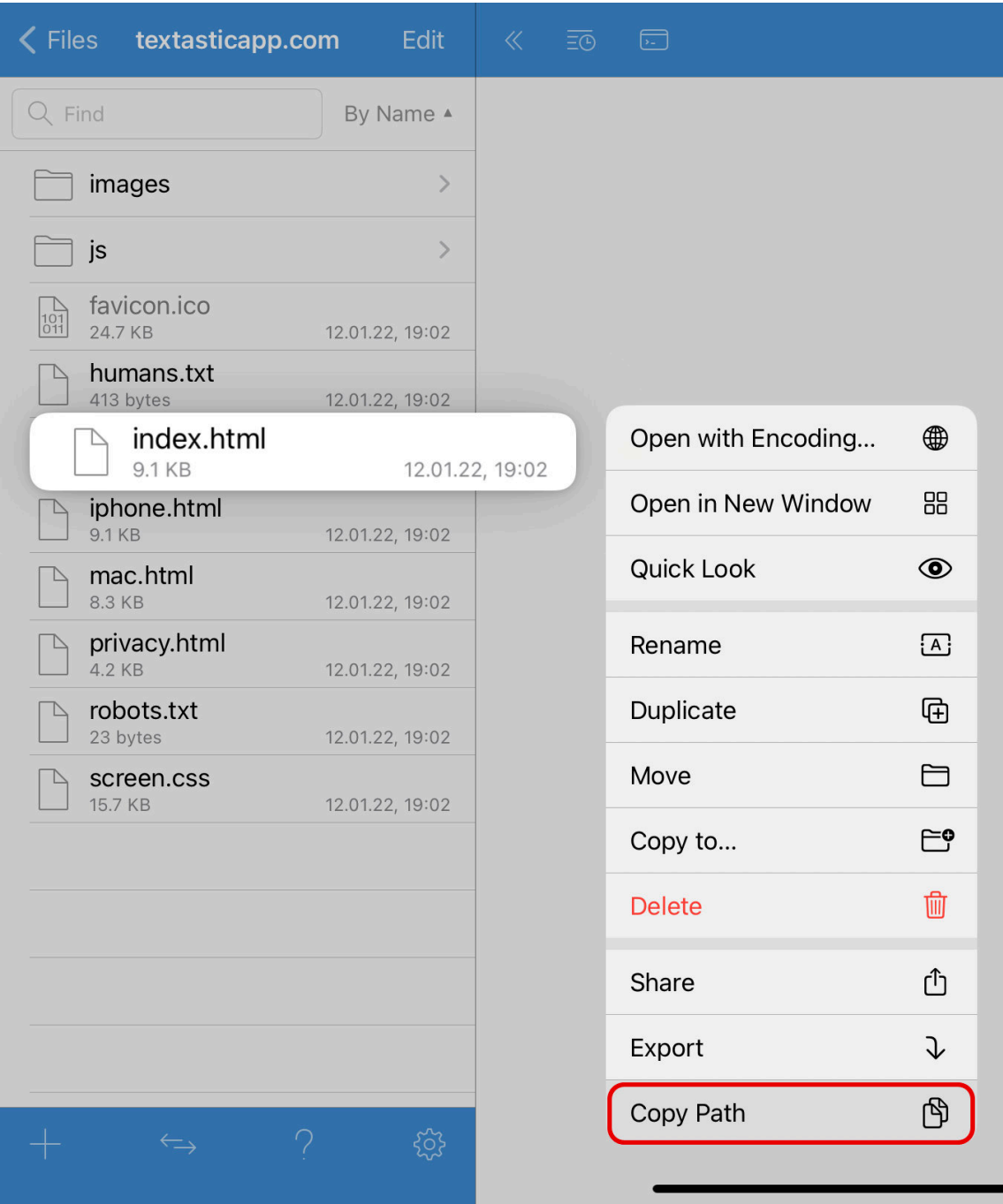
5 Viewing and Editing Files

5.1 Managing Multiple Open Files Using the Tab Bar

You can open multiple files at the same time and switch between them using the tab bar.
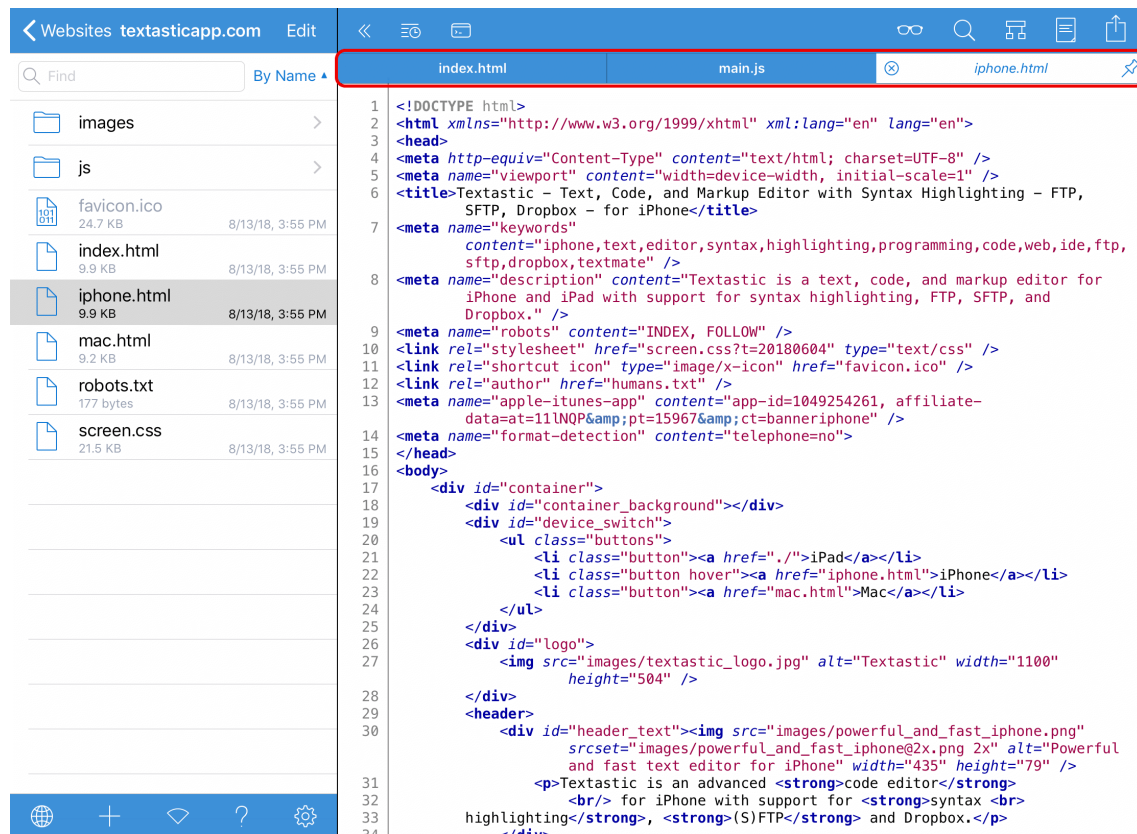
When you open a file, it is opened in a **transient** (or temporary) tab, indicated by the *italic* title and the *Pin* button on the right side of the tab.

To make the tab stay when you open another file, tap the file title or the pin button at the right side of the tab. The file is also kept open if you tap on the editor or open the same file by tapping on it a second time.

Note External files opened using the document picker and recent files are not opened in a transient tab but always stay open when you open another file.

This is very similar to what desktop code editors like Sublime Text, Visual Studio Code, Atom and TextMate do: in those apps, files are only kept open if you do double-click them. This is done so that you don't end up with many, many tabs when you are just browsing a project.

Drag tabs to **reorder** them.

5.2 Web and Markdown Preview

Textastic has a built-in web preview that can preview HTML, CSS, JavaScript and Markdown files. Tap the *Web Preview* button ("glasses" icon) to open the web preview screen.
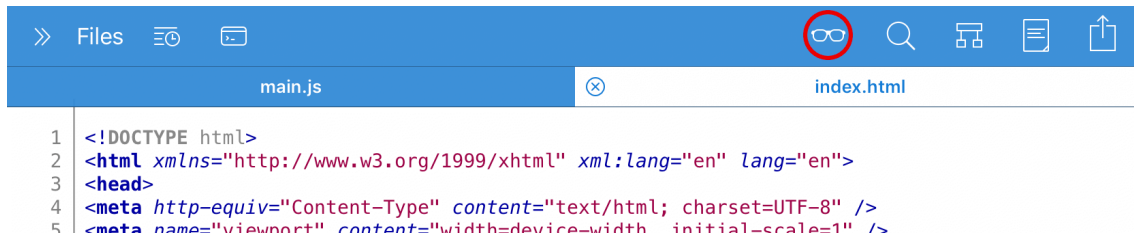
The web preview screen uses the same WebKit engine that is used by Safari and provided by iOS (WKWebView). For previewing Markdown, Textastic uses the MultiMarkdown library.

By default, tapping on the web preview button will show a preview of the file currently open in the editor.

If you want to view another local file, you can enter a different local **file path** in the address bar.

If instead, you want to view a remote file, you can switch from *Local* to *Remote* and enter an arbitrary **URL** in the address bar.

Textastic - Text, Code, and Markup Editor with Syntax Highlighting - FTP, SFTP, SSH, Dropbox, Google Drive - for iPad
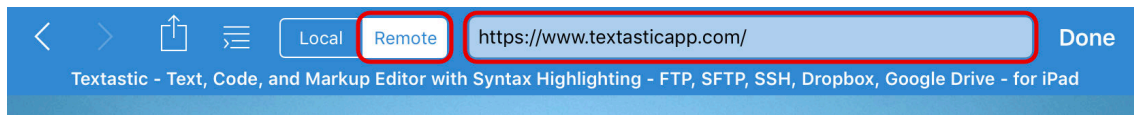
Important URLs (including query parameters) are only supported when you switch to *Remote* preview. For *Local* preview, you can only enter file paths that are relative to the root of the file you are previewing. For example, if you preview a file stored in Local Files, enter a file path relative to the Local Files root folder.

Textastic will remember the local path and remote URL you enter for each file and restore it the next time you open the web preview for that file.

To **reset** the local path or remote URL, tap on the address bar, clear the contents of the address bar using the gray *X* button and press *Go* or `Return` on the keyboard.

Note Textastic offers syntax highlighting for many languages, but it doesn't include compilers or interpreters.

If, for example, you try to preview a **PHP** file locally, only the HTML part will be interpreted by the web view. *The PHP part will be ignored.*

In order to preview PHP in Textastic, you can upload the file to a test server, open the preview screen, switch from *Local* to *Remote* and enter the remote URL of your test server in the address bar.

This can also be used when editing **LaTeX** files: you could for example upload the `.tex` file to a server using SFTP, use the SSH Terminal to compile the file into a **PDF** on the server and copy it to a web server location on the server. Then you can switch from *Local* to *Remote* and enter the URL of the PDF file.

5.2.1 JavaScript Console

For debugging purposes, you can use the JavaScript console. It shows JavaScript errors and messages logged using the JavaScript `Console` object. If available, the file and line number that caused the message is also shown.
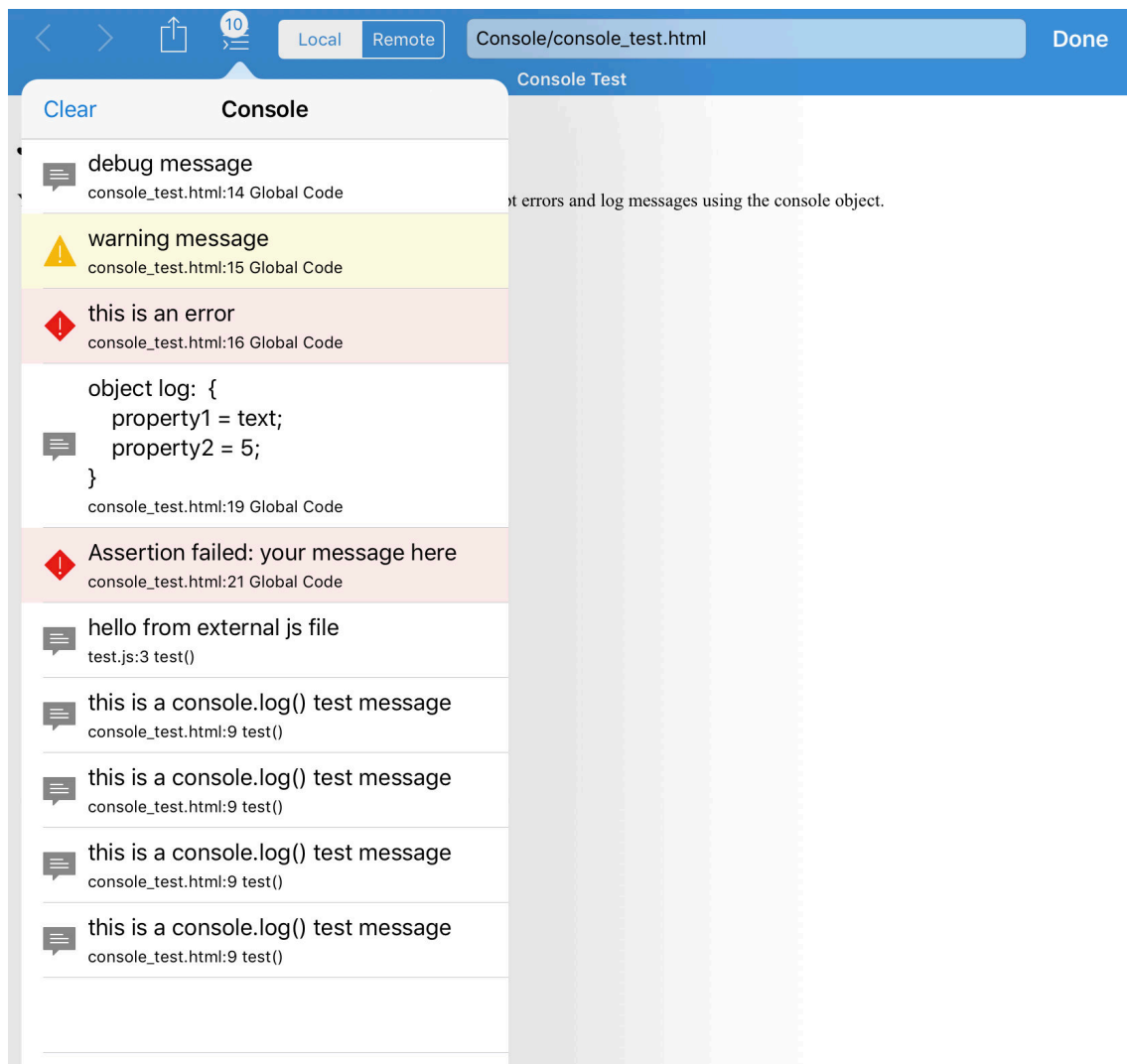
If there are any messages in the console, the *Console* button ("list with arrow" icon) displays the number of messages in a badge.

The messages in the screenshot above were generated by the following HTML file with JavaScript code:

Listing 5.1. console_test.html

```html
<!doctype html>
<html lang="en">
<head>
        <meta charset="UTF-8" />
        <title>Console Test</title>

        <script>
                function test() {
                        console.log("this is a console.log() test message");
                }

                window.setInterval(test, 1000);

                console.debug("debug message");
                console.warn("warning message");
```

```
                    console.error("this is an error");

                    var obj = { property1: "text", property2: 5 };
                    console.log("object log: ", obj);

                    console.assert(obj.stri, "your message here")
        </script>
        <script src="test.js" type="text/javascript"></script>

</head>
<body>

<div id="main">

  <h1>JavaScript Console Test File</h1>

  <p>You can use Textastic's built-in JavaScript console to see JavaScript
errors and log messages using the console object.</p>

</div>
</body>
</html>
```

Listing 5.2. test.js

```
function test() {
        console.log("hello from external js file");
}

test();
```

You can tap on a message in the console to **copy the message** or to **open the source file** at the location that caused the message:
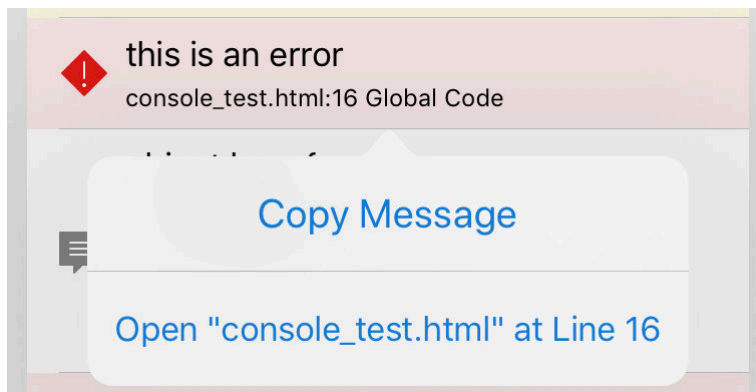
Note You cannot preview a JavaScript file directly. It must always be contained in or referenced by an HTML file.

5.2.2 Preview in Safari

When previewing a file, Textastic runs a small web server that serves static files. This enables another powerful feature: opening the preview in Safari.

On iPad, you can use that in combination with **Split View** to see the **source code of a website in Textastic and a web preview in Safari at the same time**:
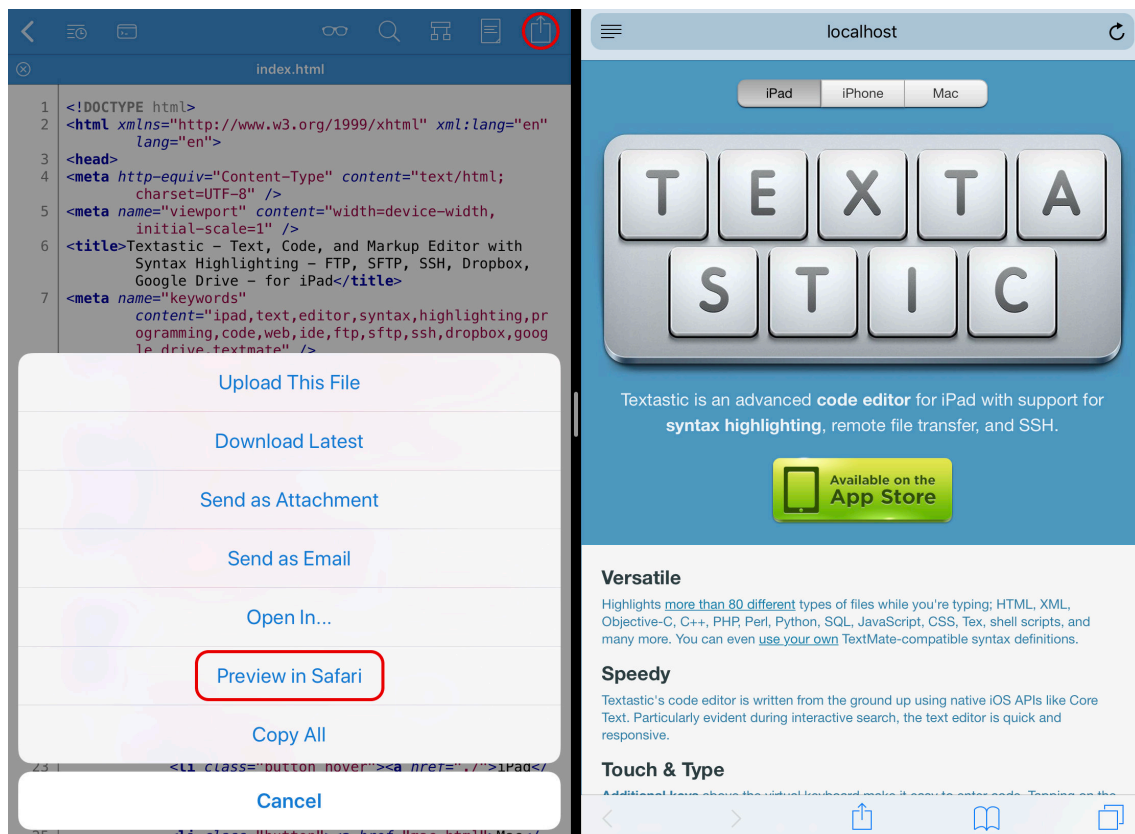
Tap on the *Action* button in the top-right corner and then on *Preview in Safari* to open the preview in a new Safari tab.

You can also tap and hold the *Web Preview* button ("glasses" icon) and choose *Preview in Safari*:
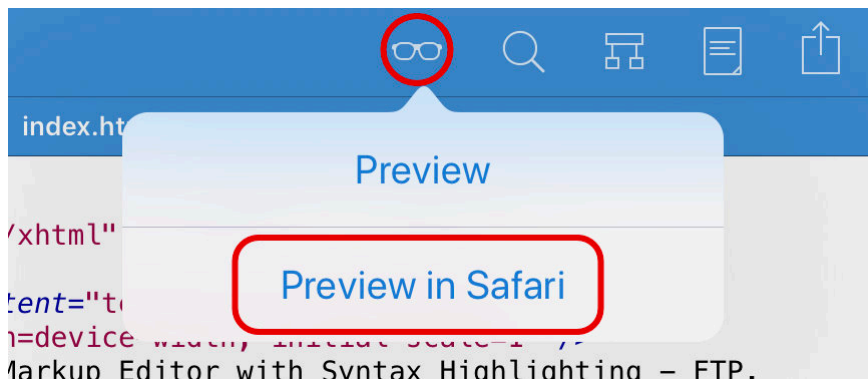
5.3 Symbol List

For many programming and markup languages, Textastic can show a list of "symbols". Depending on the file type, these can be for example class and function names, headings, CSS selectors or HTML id attributes.

Tap the *Symbols* button to show the list of symbols.

This is how it looks for an Objective-C file:

Tap on any symbol to scroll to and highlight the place where the symbol is defined.

You can also search the list:
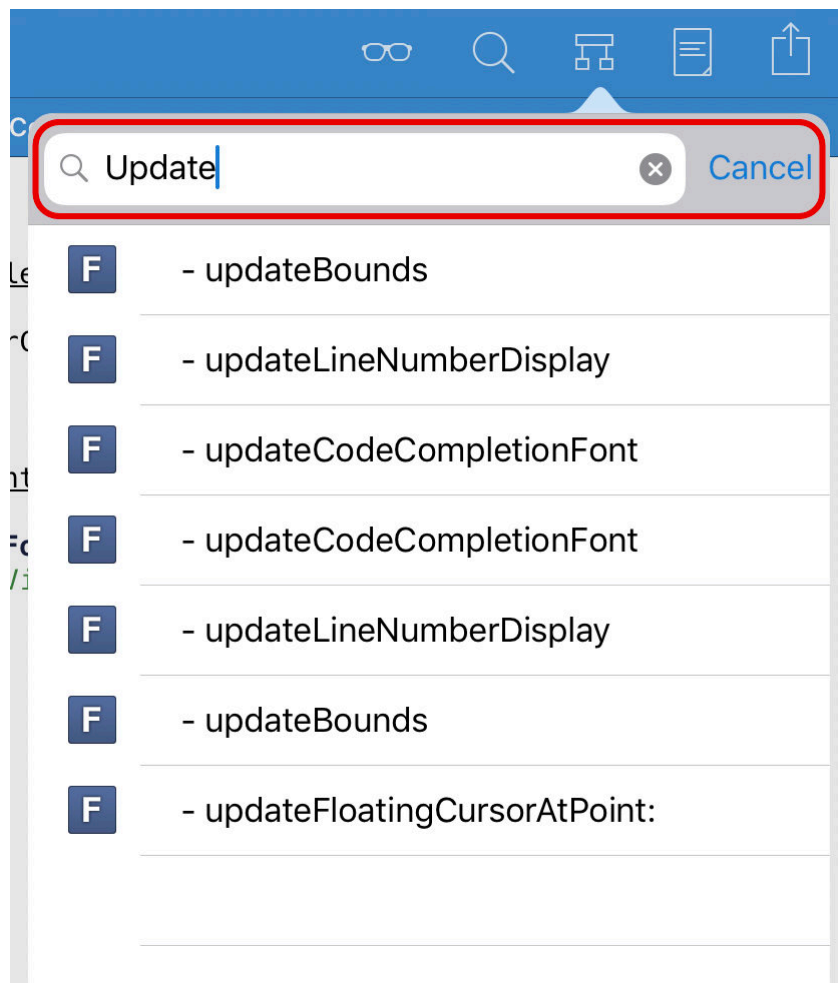
If you have a hardware keyboard connected, you can use the cursor keys to navigate in the symbol list.

5.4 Searching Within a File

Tap the *Search* button ("loupe" icon) to open the find and replace screen.
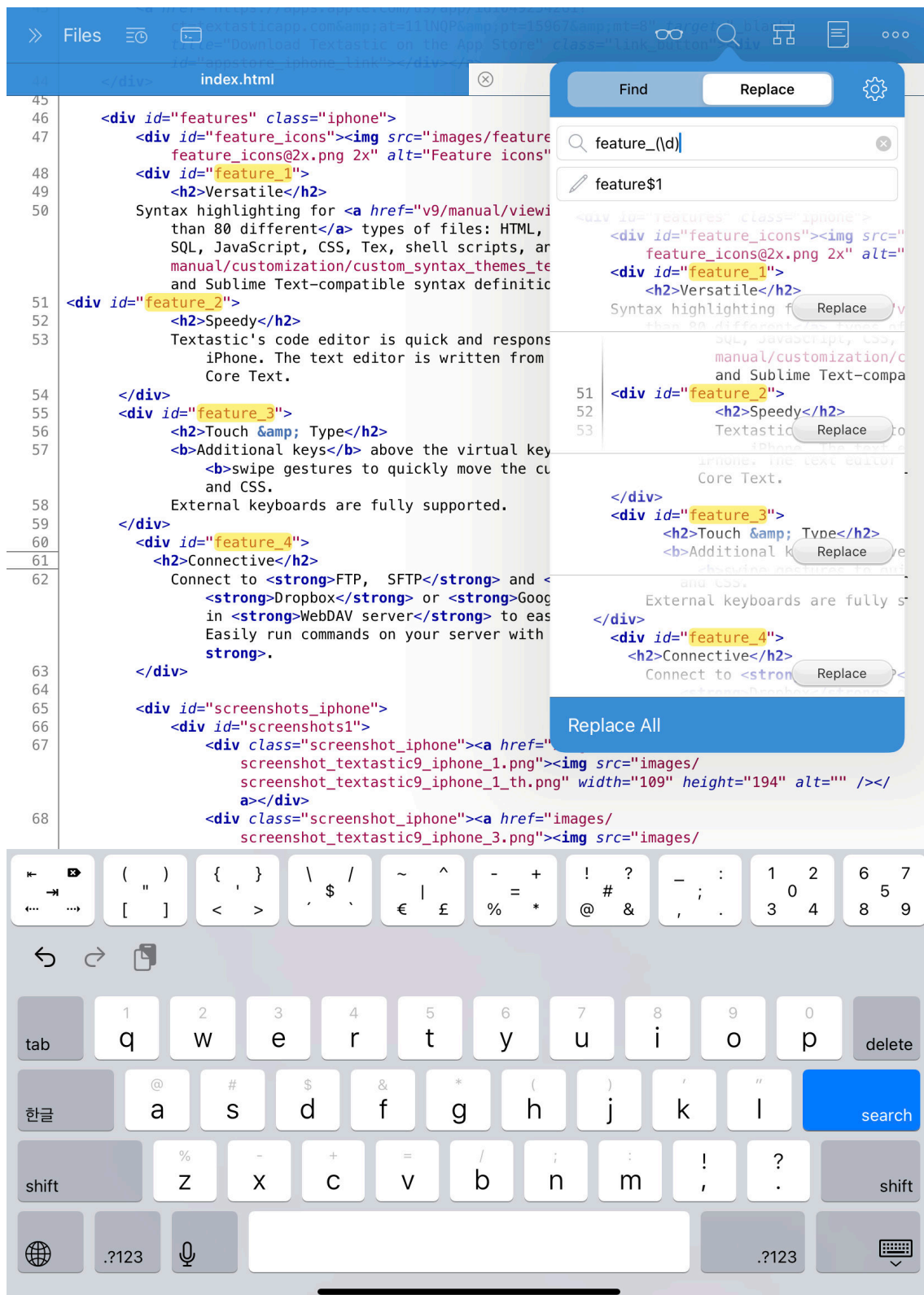
Tap *Find* or *Replace* to switch between find and replace modes.

After entering text to search for in the find text field, Textastic will immediately display a list of search results highlighted in yellow and will also highlighted the text in the code editor.

Tap a search result to scroll to and highlight the place where the search term was found.

See also

Use Find in Files to search in multiple files at once.

5.4.1 Search Settings

Tap the *Settings* button ("gear" icon) to change search settings:
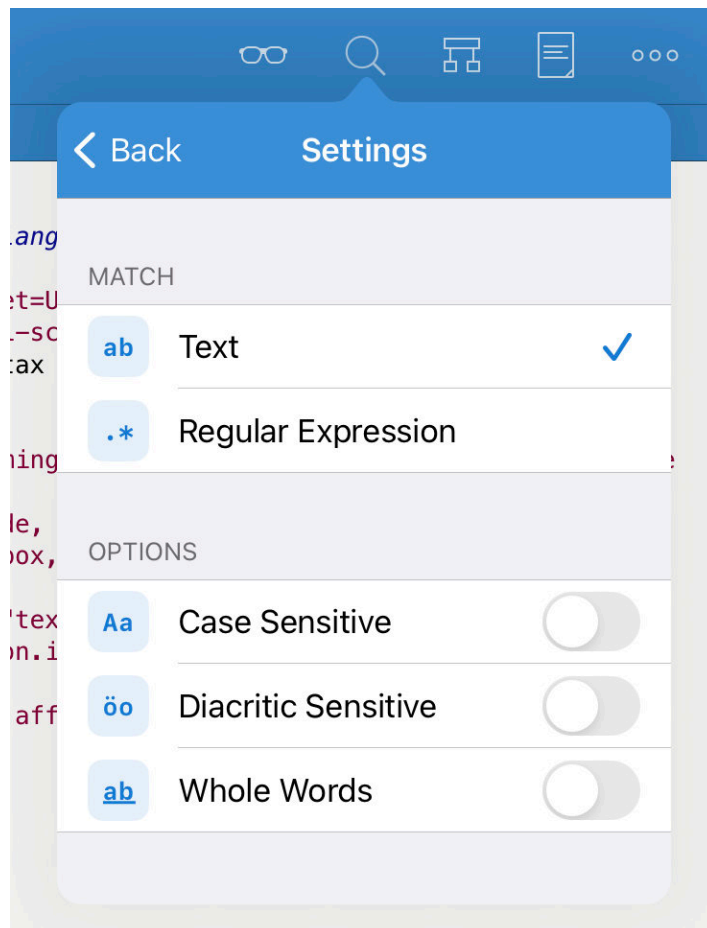
Text

Search for a simple text string. You can also search for "\n" or "\r" to find newline characters and "\t" to find tab characters. Enter "\\" to search for a backslash.

Regular Expression

Enable regular expression search. Textastic uses the ICU regular expression syntax and supports TextMate's replacement string syntax including capture references, case foldings and conditional insertions.

Use $n to reference a capture in the replace string, where *n* is the capture register number. Using $0 means the entire match.

Case Sensitive

By default, search is case-insensitive. Enable this setting to make search case sensitive and only find instances with matching case.

Diacritic Sensitive

By default, search ignores diacritic marks: for example, 'ö' is equal to 'o' and 'é' is equal to

'e'. Enable this setting to only find letters with matching diacritic marks.

Note This setting is not shown when using regular expression search. Regular expression search is always diacritic sensitive.

Whole Words

If you enable this setting, the search will only find whole words. For example, a search for "word" will find "word" but **not** "wording" or "password".

5.4.2 Replace

Tap the *Replace* button in a search result to replace that search result using the value entered in the replace text field.

Tap the *Replace All* button to replace all search results.

See Section 5.4.1 for details about the supported replace string syntax when using regular expression search.

5.5 File Properties

Tap the *File Properties* button ("document" icon) to view the file properties.

File Properties   Edit

Filename   index.html >

LINKS TO REMOTE CONNECTIONS

Dropbox Connection /Websites/textasti...

textasticapp.com /textasticapp.com/in...

FILE ATTRIBUTES

Location   Local Files/textasticapp.com

File Size   9.1 KB

Modified   14. Jan 2022 at 18:13:34

TEXT PROPERTIES

Encoding   Unicode (UTF-8) >

Line Endings   Unix (LF) >

Syntax Definition   HTML >

Line Count   139

Character Count   9185

Word Count   699

Tap the *Filename* to **rename** the file.

The *Links to Remote Connections* section shows all remote connections and file paths from where the file was downloaded or to which it was uploaded to. Have a look at Quickly Uploading and Downloading Files for more information.

The *File Attributes* section shows the file's **location**, **size** and the date it was last **modified**.

In the *Text Properties* section, you can see the current **text encoding**, **line endings**, and **syntax definition**. Tap one of these three settings to change it.

The newly selected encoding and line ending will be used the next time the file is (auto-)saved. If instead you want to open a file with a different encoding, please have a look at Opening a File with a Different Character Encoding.

In the last section, you'll find some statistics about the file: **line count**, **character count**, and **word count**.

5.6 Additional Row of Keys Above the Standard Keyboard

Textastic adds a set of additional keys above the standard on-screen keyboard. These allows you to enter up to 50 different special characters quickly without having to switch to another keyboard layout or scrolling a list of keys.
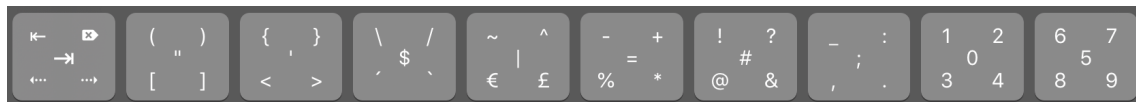
Figure 5.1. Keyboard row on iPad

Figure 5.2. Keyboard row on iPhone

5.6.1 Using the Swipeable Keys

If you want to enter the character in the middle of the button, just tap the button. If you want to enter one of the other four characters, make sure to **swipe over the button in the direction of the character** you want to type.

5.6.2 Tab Key

The first key in the row has some special functions:

Tap the key to enter a **tab character**. If text is selected, pressing the tab key will **indent the selected lines**. This is equivalent to *Shift Right* in the editing menu.

The character in the top left corner is an **"untab" character**. It has the reverse effect of the tab character and allows you to unindent selected lines. This is equivalent to *Shift Left* in the editing

menu.

The character in the top right corner is **forward delete**. It deletes the character that follows after the cursor and is the equivalent of the `Del` key on regular keyboards.

The bottom left character moves the cursor to the **beginning of the current line**. The bottom right character moves the cursor to the **end of the current line**.

5.6.3 Undo, Redo and Hide Keyboard on iPhone
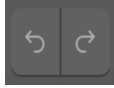
Tap the arrow pointing to the left to undo your last editing step. Tap the arrow pointing to the right for redo.

The right-most button can be used to hide the keyboard.

5.7 External Keyboard Shortcuts

Here you can find a list of keyboard shortcuts you can use when you connect an external keyboard to your iPad or iPhone.

On iPad, you can press and hold `Cmd` to see a list of shortcuts supported in the current state of the app:
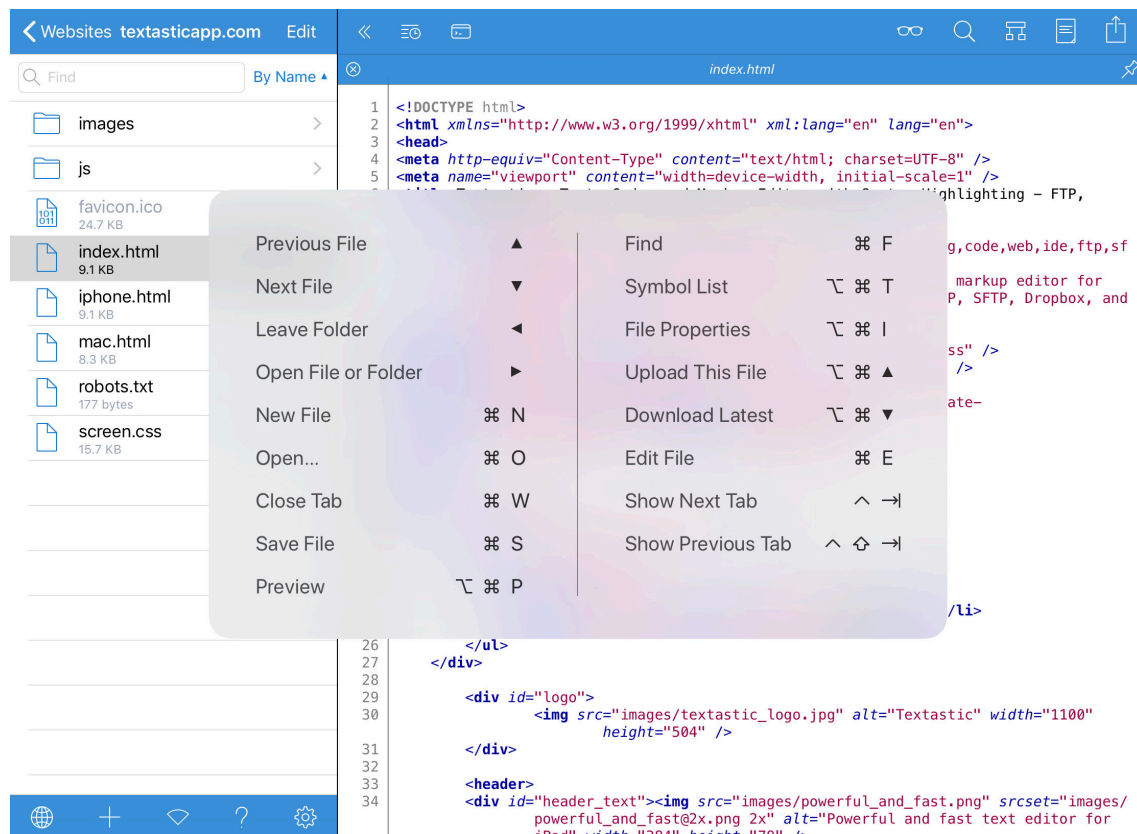
5.7.1 Keyboard Shortcuts Supported by iOS

Keyboard Shortcut	Description
`Cmd-A`	Select All
`Cmd-C`	Copy
`Cmd-X`	Cut
`Cmd-V`	Paste
`Cmd-Z`	Undo
`Cmd-Shift-Z`	Redo
`Shift-Arrow Keys`	Select text
`Cmd-Up/Down`	Move cursor to the beginning/end of the file
`Cmd-Left/Right`	Move cursor to beginning/end of the current line
`Opt-Left/Right`	Move cursor to the previous/next word
`Esc` or `Eject`	Bring up virtual on-screen keyboard
`Opt-Delete`	Delete the word to the left of the cursor
`Cmd-Delete`	Delete everything on the current line to the left of the cursor

5.7.2 Textastic's Keyboard Shortcuts

Keyboard Shortcut	Description
`Cmd-N`	Create new file/folder
`Cmd-O`	Open… (invoke iOS document picker)
`Cmd-S`	Save file
`Cmd-E`	Edit currently open file
`Cmd-L`	Go to line
`Cmd-P`	Print
`Cmd-F`	Toggle Find
`Opt-Cmd-F`	Toggle Find and Replace
`Cmd-G`	Find next
`Cmd-Shift-G`	Find previous
`Cmd-W`	Close tab
`Opt-Cmd-B`	Toggle sidebar
`Opt-Cmd-P`	Toggle preview
`Ctrl-Opt-Cmd-P`	Preview in Safari
`Opt-Cmd-I`	Toggle file properties popover
`Opt-Cmd-T`	Toggle symbol list popover
`Opt-Cmd-Up`	Upload current file
`Opt-Cmd-Down`	Download current file
`Tab`	Indent selected text
`Shift-Tab`	Unindent selected text
`Ctrl-Tab`	Select next tab
`Ctrl-Shift-Tab`	Select previous tab
`Cmd-[`	Shift left
`Cmd-]`	Shift right
`Ctrl-E`	Expand Emmet abbreviation
`Ctrl-A`	Wrap with Emmet abbreviation
`Cmd-I`	Toggle italics (works in Markdown and HTML files)

`Cmd-B`	Toggle boldface (works in Markdown and HTML files)
`Cmd-U`	Toggle underline (works in HTML files)
`Esc,Cmd-.,Cmd-W`	Close dialog/popover
`Arrow Keys`	Navigate in the list of files, search results, and in the symbol list

Important You can configure all keyboard shortcuts in Textastic's settings.

5.7.3 Emacs Key Bindings supported by iOS

Keyboard Shortcut	Description
`Ctrl-A`	Move cursor to the beginning of the line
`Ctrl-E`	Move cursor to the end of the line
`Ctrl-F`	Move cursor forward one character
`Ctrl-B`	Move cursor backward one character
`Ctrl-K`	Kill/delete text between cursor and the end of the line
`Ctrl-H`	Delete one character to the left

5.8 Cursor Navigation Wheel

On iPad, Textastic has a special control that allows you to quickly change the current selection.

You can bring up the cursor navigation wheel **by tapping on the editor with two fingers**.
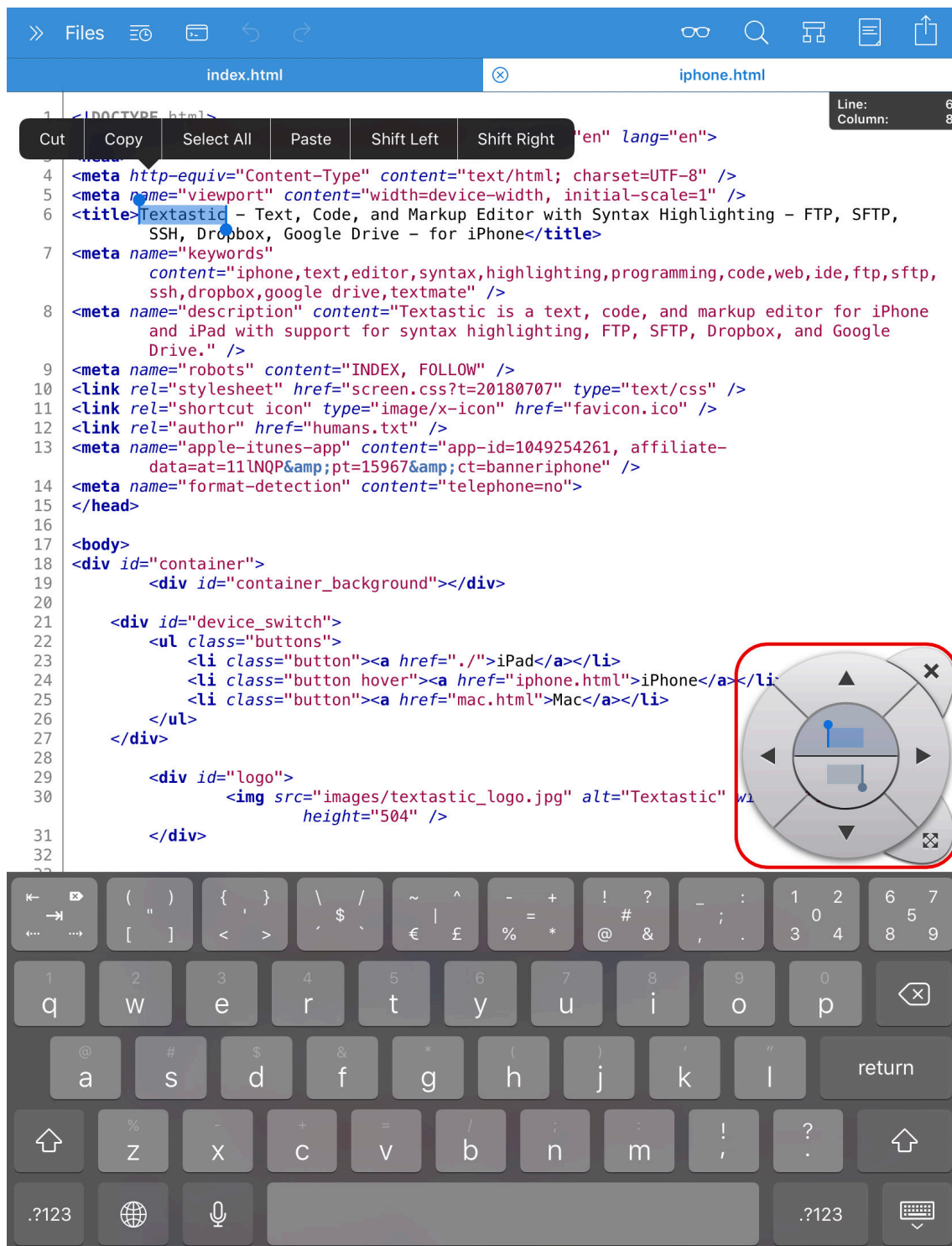
You can close the control by tapping with two fingers again or by using the close button on the top right.

Move the control by using the button on the lower right.

5.9 Opening a File with a Different Character Encoding

If Textastic does not automatically detect the correct text encoding when opening a file, you can manually select a different character encoding.

- Tap and hold the file
- Tap *Open with Encoding*
- Select a character encoding
- Tap *Done*

favicon.ico	24.7 KB	8/30/18, 5:54 PM	

Cancel	Choose Character Encoding	Done

Unicode (UTF-8)

Western (ASCII)

Western (ISO Latin 1) ✓

Western (Mac OS Roman)

Western (Windows Latin 1, CP 1252)

Western (DOS Latin 1, CP 850)

Western (ISO Latin 3)

Western (ISO Latin 9)

Latin-US (DOS, CP 437)

Baltic (ISO Latin 7)

Baltic (Windows)

Baltic (DOS)

Central European (ISO Latin 2)
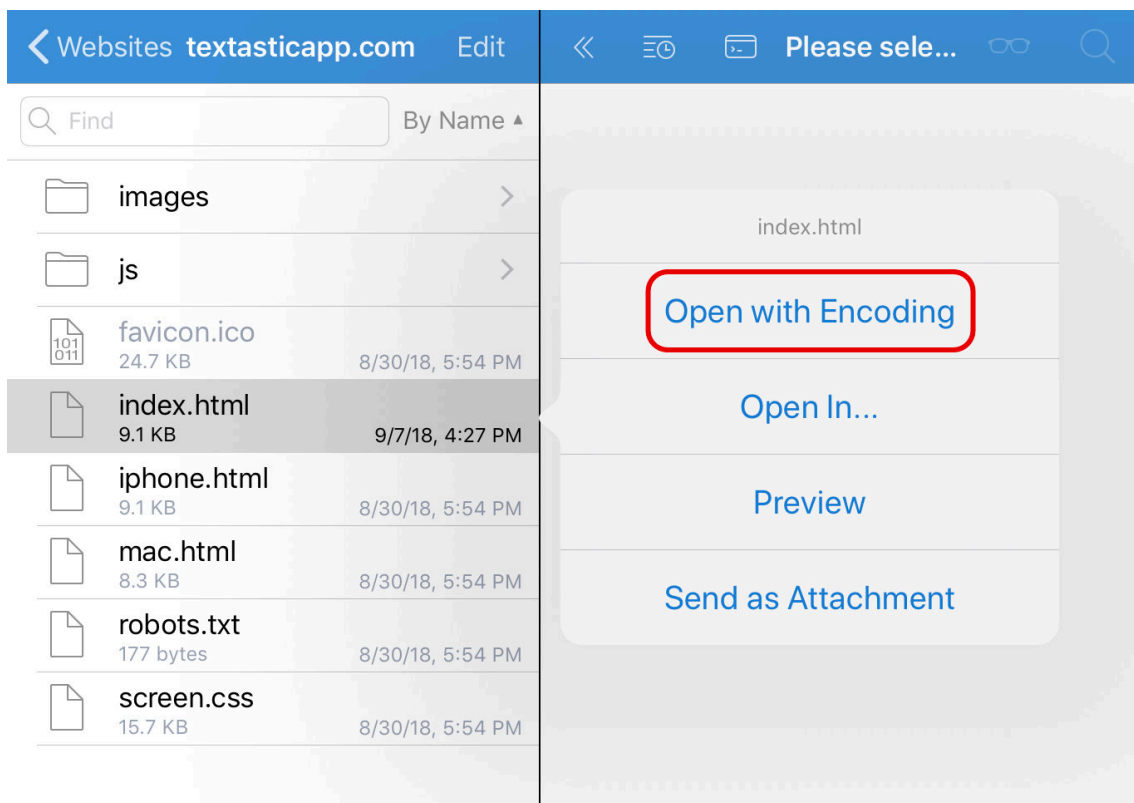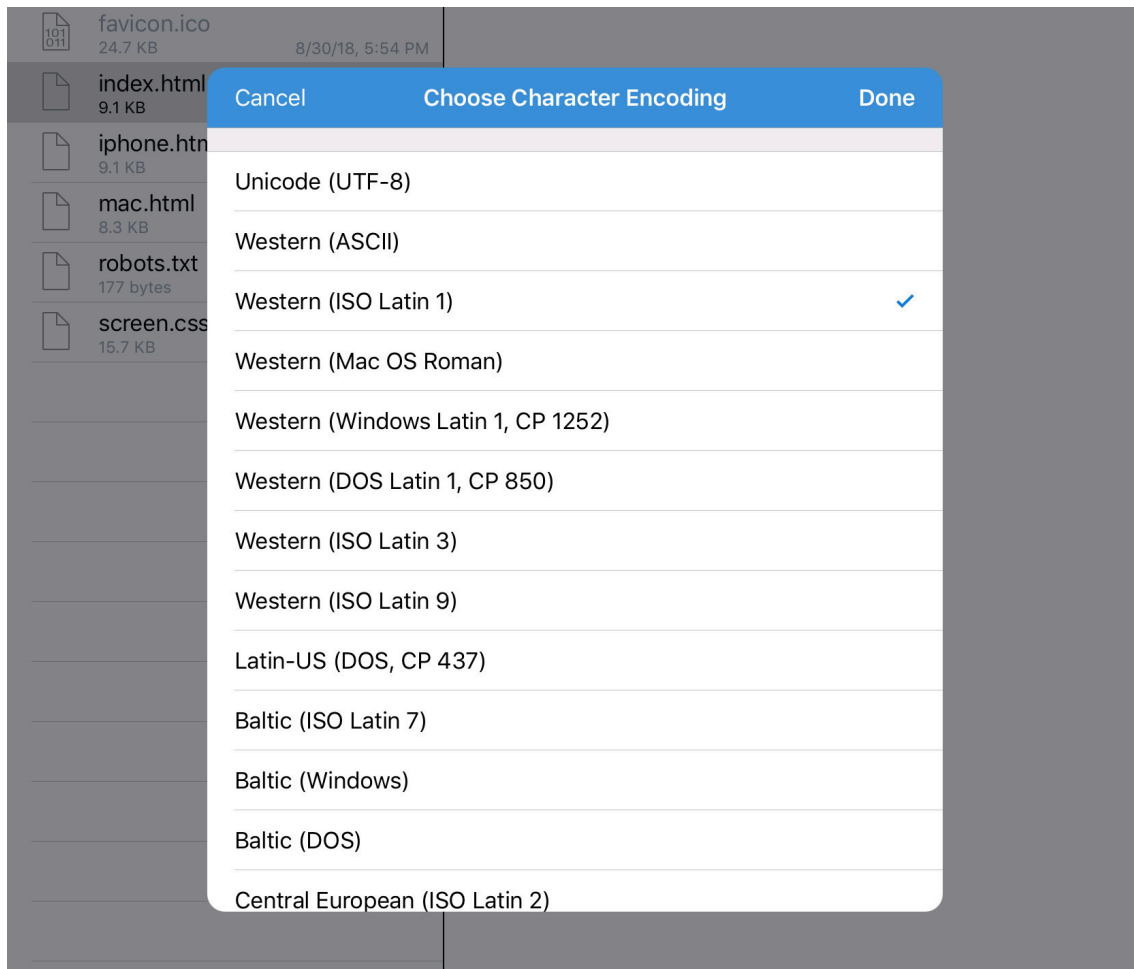
After the file is opened with the newly selected character encoding, you will be asked if you want to save the file with the new character encoding. Only choose *Save* if the file is displayed correctly. Otherwise tap *Close File*.

Save With Selected Character Encoding?

Would you like to save the file with the newly selected character encoding "Western (ISO Latin 1)"?

You should only save the newly selected character encoding if the file is displayed correctly.

Close File   Save
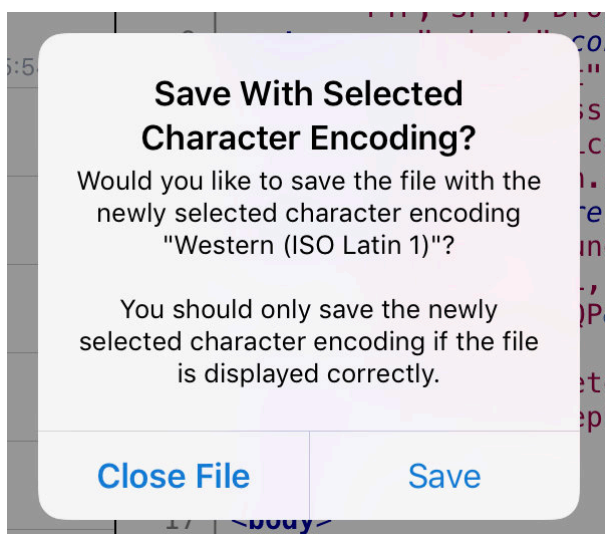
5.10 Supported File Types

Textastic supports syntax highlighting of the following plain text file types:

Note Popular programming and markup languages are highlighted with **bold** text.

- ActionScript
- Active4D
- Ada
- Apache (conf, htaccess)
- AppleScript
- Arduino
- ASP, ASP vb.NET
- Batch files
- BibTeX
- **C**
- **C#**
- **C++**
- Clojure
- CoffeeScript
- ColdFusion
- **CSS**
- CSV
- D
- Diff
- DokuWiki
- Dylan
- Eiffel
- Erlang
- Fortran
- Fountain
- **Go**
- Grails Server Page
- Groovy
- Haskell
- **HTML** (various variants)
- Ini
- **Java**
- Java properties

- Java Server Page
- **JavaScript**
- JSON
- **LaTeX**
- LESS
- LilyPond
- Linden Scripting Language (LSL)
- Lisp
- LiveCode
- Logo
- Lua
- Mail
- Makefile
- **Markdown**
- MATLAB
- Mediawiki
- MEL
- MIPS Assembler
- Modula-3
- MulitMarkdown
- Nemerle
- **Objective-C**
- Objective-C++
- OCaml
- Octave
- Pascal
- **Perl**
- **PHP**
- Plain Text
- PowerShell
- Prolog
- Property List
- **Python**
- R
- Regular Expressions
- RJS
- reStructuredText
- **Ruby**
- Ruby Haml

- Ruby on Rails
- **Rust**
- Sass
- SCSS
- Scala
- Scheme
- Shell Script (Bash)
- SQL
- SSH Config
- Stata
- Strings File
- **Swift**
- TCL
- TeX
- Textile
- TSV
- **TypeScript**
- Verilog
- VHDL
- **XML**
- XSL
- **YAML**

Hint Textastic uses syntax definitions and themes from Sublime Text 3 and TextMate 1. You can also add your own syntax definitions and themes.

6 Settings

6 Settings

6.1 Overview

Tap the *Settings* button ("gear" icon) in the bottom toolbar to open the settings overview.
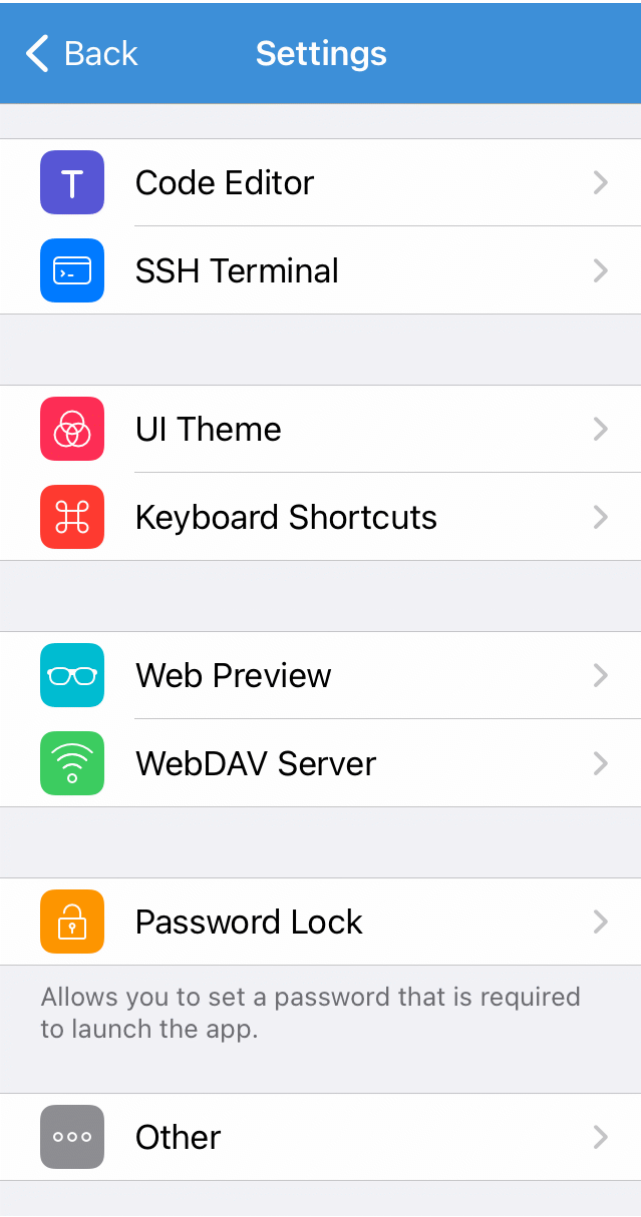
6.2 Code Editor

6.2.1 Font and Line Height

Font	Menlo >	
Font Size	13 pt	− +
Line Height	110 %	− +

Font
  Choose from a selection of built-in monospaced fonts. You can also install custom fonts.

Font Size
  Increase or decrease the editor font size using the stepper.

Line Height
  Increase or decrease the line height. Use it to adjust the spacing between lines in the code editor. The default is 110%.

6.2.2 Editor Theme and Keyboard Appearance

LIGHT APPEARANCE	
Theme	Vintage >
Dark Keyboard	⬜

DARK APPEARANCE	
Theme	Solarized (dark) >
Dark Keyboard	🔵

Select a separate theme and keyboard appearance for Dark Mode.

Since iOS 13, users can choose to adopt a dark system-wide appearance called **Dark Mode**.

In Textastic, you can set separate themes and keyboard appearances for *light appearance* and *dark appearance*.

Theme
  Select the theme you want to use in the code editor. The list of themes starts with light themes and ends with dark themes. You can also add custom themes.

Dark Keyboard

Enable this setting if you want to use a dark keyboard instead of the default light keyboard.

See also

Have a look at UI Theme on how to select the UI theme to use for dark and light system appearances.

6.2.3 Gutter and Line Numbers
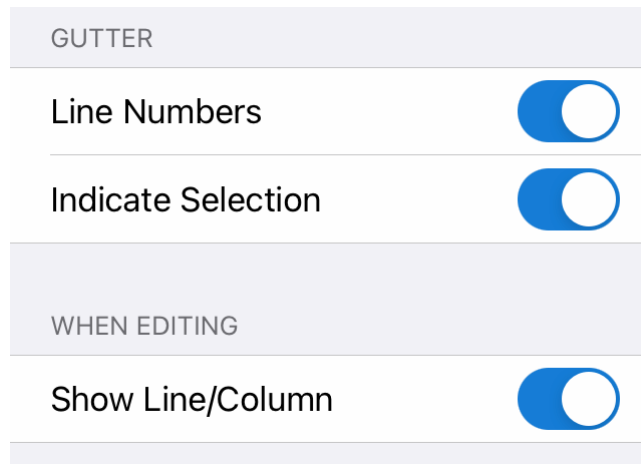
Line Numbers

Enable this settings to display the number of each line on the left side of each line.

Indicate Selection

If line numbers are enabled, you can choose to indicate the start and end of the current selection in the gutter:
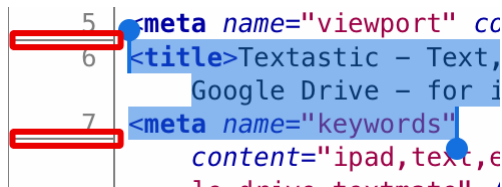
Show Line/Column

Show a line/column indicator on the right side above the code editor while editing a file (*iPad only*).
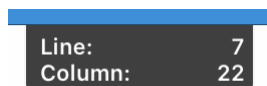
6.2.4 Tab Characters

TAB CHARACTERS

| Tab Width | 4 | − | + |
| Soft Tabs | | | ⬜ |

Enable "Soft Tabs" to insert spaces instead of a tab character when you press the Tab key.

Tab Width
Select the number of spaces one tab corresponds to.

Soft Tabs
Enable to insert spaces instead of a tab character when you press the `Tab` key.

6.2.5 Invisible Characters

INVISIBLE CHARACTERS

Show Tabs	⬜
Show Spaces	⬜
Show Newlines	⬜

Show All

Show Tabs
Enable to show tab characters as small gray triangles in the editor.

Show Spaces
Enable to show space characters as dots in the editor.

Show Newlines
Enable to indicate newline characters at the end of each line in the editor.

Show All / Show None
Tap to quickly show all or no invisible characters.

6.2.6 Word Wrap

WORD WRAP	
Indent Wrapped Lines	●━
Auto Word Wrap	━○
Wrap Column	100

Indent Wrapped Lines

    Enable to indent soft wrapped lines. This makes it much easier to read code, but you may want to disable it for regular text/prose.

Auto Word Wrap

    Enable to visually wrap lines (soft wrap) to the editor width.

Wrap Column

    If *Auto Word Wrap* is disabled, you can manually enter a column at which lines are soft-wrapped.

6.2.7 Keyboard

KEYBOARD	
Read–Only	━○
Auto-Correction	━○
Auto-Capitalization	━○
Smart Punctuation	━○
Additional Keys	●━
Shortcuts	━○

Additional keys are displayed only when no hardware keyboard is connected.

Read-Only

    Enable this settings to make the editor read-only. Tapping on the editor won't bring up the keyboard anymore.

Auto-Correction

    Enable to spellcheck words as you type, automatically correcting misspelled words. You should disable this if editing code.

Auto-Capitalization

Enable automatic capitalization of the first letter of each sentence.

Smart Punctuation

Enable to allow smart quotes and automatic conversion between hyphens and en or em dashes.

Additional Keys

Enable the additional row of keys above the standard keyboard (*iPad only*).

Shortcuts

Enable the standard shortcuts bar above the keyboard which has buttons for undo/redo and copy/cut/paste (*iPad only*). If you enable *Additional Keys*, this should be disabled to save screen space.

6.2.8 Text Editing

Code Completion

Enable code completion. Currently, Textastic supports code completion for HTML, JavaScript, CSS, C, Objective-C, and PHP.

Indent Based on Syntax

Automatically indent code when typing, based on its syntax. For example, increases the indentation when pressing Return after an opening curly brace ("{") in C or an opening tag in HTML and decreases indentation after a closing curly brace ("}") or a closing tag.

Close Brackets and Quotes

Automatically insert a matching closing character when typing an opening bracket or quote. This works for the following pairs: `""`, `''`, `()`, `[]`, `{}`.

6.2.9 Emmet

Emmet Cheat Sheet

> Open the Emmet Cheat Sheet in the browser. Also have a look at the Emmet Documentation.

Expand with Tab

> Enable to expand Emmet abbreviations with the `Tab` key if possible (this works for example in HTML, CSS, XML, Less, Sass, and SCSS files).
>
> You can also use `Ctrl-E` on an external keyboard to expand Emmet abbreviations. See External Keyboard Shortcuts.

6.2.10 Cursor Navigation Wheel

CURSOR NAVIGATION WHEEL	
Show Hints	🔵

You can show the cursor navigation wheel by tapping on the editor with two fingers.

Show Hints

> By default, the cursor navigation wheel shows hints when adding, removing or moving selection handles with the buttons in the middle of the wheel. You can disable this here.

6.2.11 TextExpander

TextExpander	🔵
Update Snippets	>

TextExpander touch is installed and snippets have been fetched.
(Last update: Today at 14:12:39)

TextExpander

> Enable TextExpander snippet expansion with this setting.

Update Snippets

> Tap here to launch the TextExpander app and fetch a copy of the latest snippets.

6.3 SSH Terminal

Font	JetBrains Mono >
Font Size	14 pt − | +
Theme	Dark >
Dark Keyboard	⬤
BELL	
Play Sound	⬤

6.3.1 Font and Font Size

Font
Choose from a selection of built-in monospaced fonts. You can also install custom fonts.

Font Size
Increase or decrease the SSH terminal font size using the stepper.

6.3.2 Terminal Theme and Keyboard Appearance

Theme
Select the theme you want to use in the SSH terminal.

Dark Keyboard
Enable this setting if you want to use a dark keyboard instead of the light keyboard.

6.3.3 Bell

Play Sound
Enable if you want to play a sound in addition to a visual indicator when the terminal receives a bell code.

6.4 UI Theme

APPEARANCE

System ✓

Light

Dark

LIGHT THEME

Blue ✓

Gray

DARK THEME

Blue ✓

Yellow

Gray

Since iOS 13, users can choose to adopt a dark system-wide appearance called **Dark Mode**.

In Textastic, you can set separate themes and keyboard appearances for *light appearance* and *dark appearance*.

6.4.1 Appearance

System
 Automatically switches to light or dark appearance according to system settings.

Light
 Always use the light appearance.

Dark
 Always use the dark appearance.

6.4.2 Light Theme

The default light UI theme is *Blue*. You can also switch to the *Gray* UI theme.

6.4.3 Dark Theme

The default dark UI theme is *Blue*. You can also switch to the *Yellow* or *Gray* UI theme.

6.5 Keyboard Shortcuts

6.5.1 List of Keyboard Shortcuts

Textastic allows you to configure the keyboard shortcuts of all supported keyboard commands.

When you open the keyboard shortcuts settings, you will see a list of all configurable keyboard commands:

‹ Back **Keyboard Shortcuts** Reset	
FILE	
New File	⌘ N
Open...	⌘ O
Save File	⌘ S
Edit File	⌘ E
Go to Line	⌘ L
Print	⌘ P
Upload This File	⌥ ⌘ ▲
Download Latest	⌥ ⌘ ▼
TAB	
Close Tab	⌘ W
Show Next Tab	⌃ ⇥
Show Previous Tab	⌃ ⇧ ⇥
SHOW	
Preview	⌥ ⌘ P
Preview in Safari	⌃ ⌥ ⌘ P
Symbol List	⌥ ⌘ T

On the right side of each command, you can see the assigned key combination. The symbols for the modifier keys follow iOS and macOS conventions and have the following meaning:

Symbol	Key
⌘	Command (or Cmd)
⌥	Option (or Alt)
^	Control (or Ctrl)
⇧	Shift

Hint On keyboards made for Windows PCs, use the Alt key instead of Option, and the Windows logo key instead of Command.

Press the *Reset* button in the navigation bar to reset all shortcuts to their default key combinations.

6.5.2 Change Keyboard Shortcut

Tap a keyboard command to configure the keyboard shortcut.

〈 Back   **Change Shortcut**

"New File"

⌘ N

Please enter the desired key combination on an external keyboard.
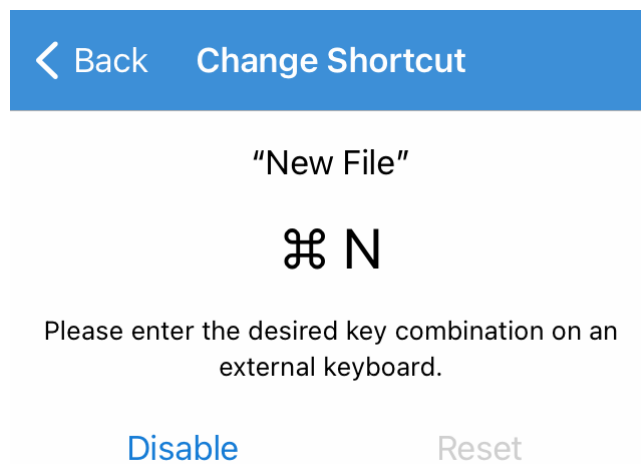
Disable   Reset

To change the shortcut, you need to have a hardware keyboard connected.

Press the desired key combination on the external keyboard.

You can also *Disable* the keyboard shortcut or *Reset* the keyboard shortcut to the default.

6.6 Web Preview

When previewing a file, Textastic runs a small web server that serves static files.

The web server for the web preview is started when the app is launched.

6.6.1 Port Settings

‹ Settings **Web Preview**	
Port	8080
Random Port	⚪

When previewing a file, Textastic runs a small web server that serves static files. You can configure the port of this web server here.

Port
> Configure the port of the web preview web server here. The default port is 8080.

Random Port
> Instead of a fixed port number, you can also let Textastic choose a random port.

Tip Use a fixed port number so you can bookmark preview URLs in Safari when using Section 5.2.2. You can change the port or use a random port in Settings -> Web Preview.

6.7 WebDAV Server

‹ Back **WebDAV Server**	
SETTINGS	
User	textastic
Password	
Port	80
Server Started	⚪
CONNECTION INFO	
WebDAV URL	http://192.168.2.122/

Here you can modifiy the settings of the WebDAV server and also start or stop the server.

6.8 Password Lock

Cancel	**Password Lock**	Done

Enable	⬤
Password	
Confirm	

The password will be requested when you launch the app. Be careful as there is no way to recover it.

Require Password	Immediately >
Face ID	⬤

You can set a password that you need to enter when launching the app. You can also enable *Touch ID* or *Face ID* (depending on your iOS device).

6.8.1 Enabling or Disabling the Password Lock

Enable
Switch this on to enable the password lock.

Password
After enabling the password lock, you need to enter the password that is required to unlock the app.

Confirm
Enter the same password again to confirm it.

Require Password
Here you can configure whether the password is required immediately after closing the app, or after 1, 5, or 15 minutes, or after 1 or 4 hours.

Face ID / Touch ID
Enable Face ID or Touch ID if you want to be able to unlock the app without having to enter the password.

After configuring the password lock, press the *Done* button to apply the settings.

Important Make sure to remember the password you set since there is no way to recover it.

> **Warning** The password lock only prevents access to Textastic itself. It can be used to protect the remote connections and SSH terminal connections you set up in the app.
>
> Even with the password lock enabled, files stored in "Local Files" are **still available in the Files app** in "On my iPad/iPhone" and can be accessed by the user once the device is unlocked.
>
> Always make sure that your device is protected with a strong passcode and only give your device to people you trust in order to protect your important files.

6.9 Other

Reload Customizations

Reloads all syntax definitions, themes, and code completions from the "Local Files/ #Textastic" folder. Have a look at the manual for further details.

Getting Started

Create file 'GettingStarted.textastic' on startup.

QUICK TRANSFER CONFIRMATION

Upload

Download

Ask for confirmation if a quick upload or download would overwrite a file?

Debug Log

Writes detailed log files to the local "logs" directory.

6.9.1 Reload Customizations

Reloads all syntax definitions, themes, and code completions from the `Local`

`Files/#Textastic` folder.

This is necessary if you want to use new custom syntax definitions, themes, and code completions.

ee Adding Custom Syntax Definitions, Themes, and Templates and Section 7.1.3 for more details.

6.9.2 Getting Started
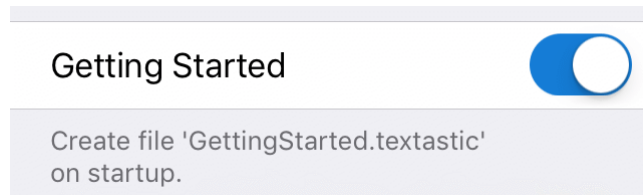
Getting Started
> By default, Textastic creates the `GettingStarted.textastic` file at launch. If you delete that file or if you manually disable this setting, the file won't be created anymore.

6.9.3 Quick Transfer Confirmation

Upload
> Enable this setting if you want to see a confirmation dialog when a quick upload would override a file on the remote server. *Enabled by default.*

Download
> Enable this setting if you want to see a confirmation dialog when a quick download would override a local file. *Enabled by default.*

6.9.4 Debug Log

Debug Log
> Enable this setting to write detailed log files to the `logs` directory in Local Files.

7 Customization

7.1 Adding Custom Syntax Definitions, Themes, and Templates

Textastic allows you to add your own custom syntax definitions and themes. You can add Sublime Text 3 packages, TextMate 1-compatible **.tmbundle** packages, and **.tmTheme** theme files.

You can also add custom file templates that appear in the file creation screen.

7.1.1 Add the Special Folder Called "#Textastic"

- Go to the Local Files root folder
- Tap the + button in the bottom toolbar
- Choose *Folder* and enter `#Textastic` as the *Folder Name*
- Tap *Create*
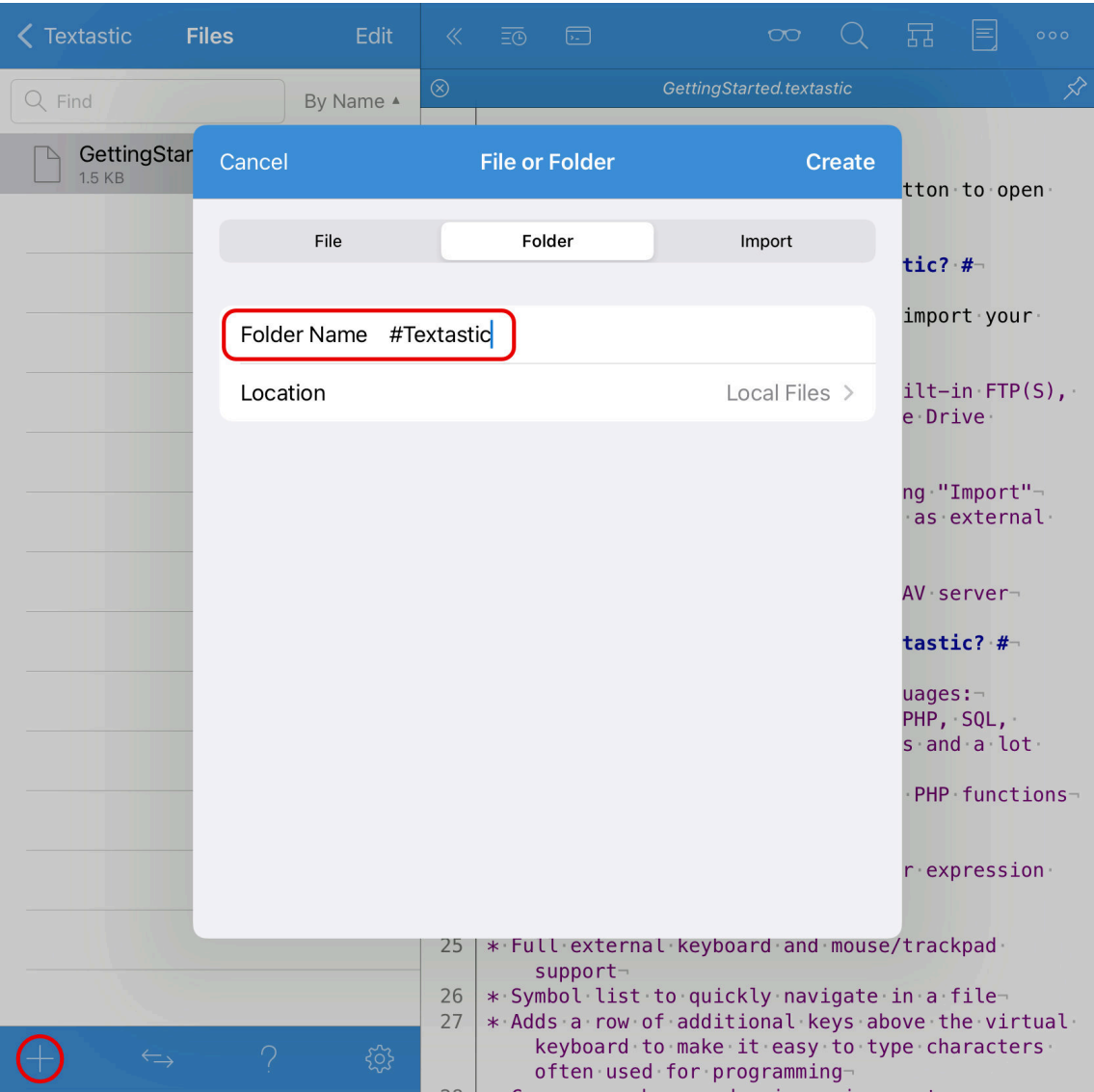
Important Make sure the newly created folder has a special icon as seen in the screenshot below. The name of the folder ("#Textastic") is **case sensitive**.
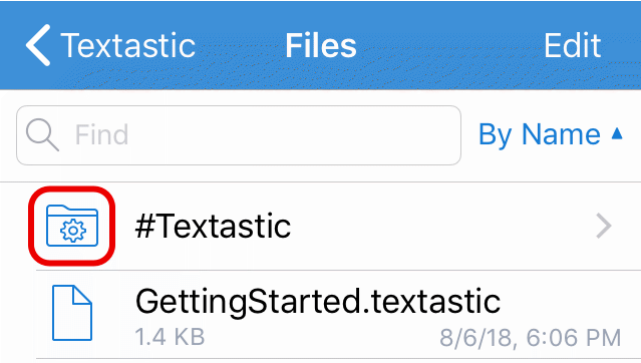
7.1.2 Add Custom Syntax Definitions and Themes

Move your Sublime Text 3 packages, TextMate bundles and theme files into the newly created directory (see Moving, Copying, and Renaming Files and Folders).
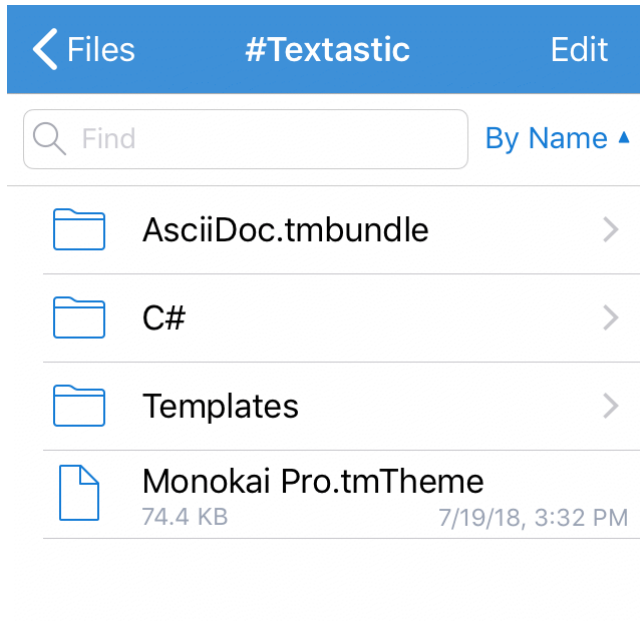
- Sublime Text 3 packages are simple folders without a special file extension.
- TextMate bundles are folders with a `.tmbundle` file extension.
- Theme files have a `.tmTheme` file extension.
- Templates have the file extension `.json`.

7.1.3 Reload Customizations

To use new custom syntax definitions, themes, and code completions, you need to reload customizations:

- Tap the *Settings* button ("gear" icon) in the bottom toolbar to open the settings screen.
- Tap *Other*.
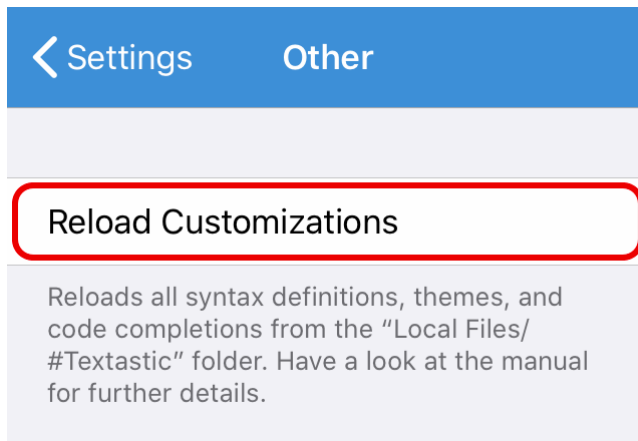
- Tap *Reload Customizations* to reload all syntax definitions, themes, and code completions.

All open files in the "Local Files/#Textastic" folder are automatically saved before reloading customizations.

Hint You can also invoke "Reload Customizations" using the customizable keyboard shortcut `Cmd-Shift-R`.

This means that you can, for example, edit a theme or code completion file and simply press `Cmd-Shift-R` on an external keyboard to save and reload it, enabling easy customization.

After reload, the new syntax definitions can be chosen in the File Properties. You can select added themes in the code editor settings.

7.1.4 Adding Custom Templates

You can also add your own templates that you can select in the file creation dialog.
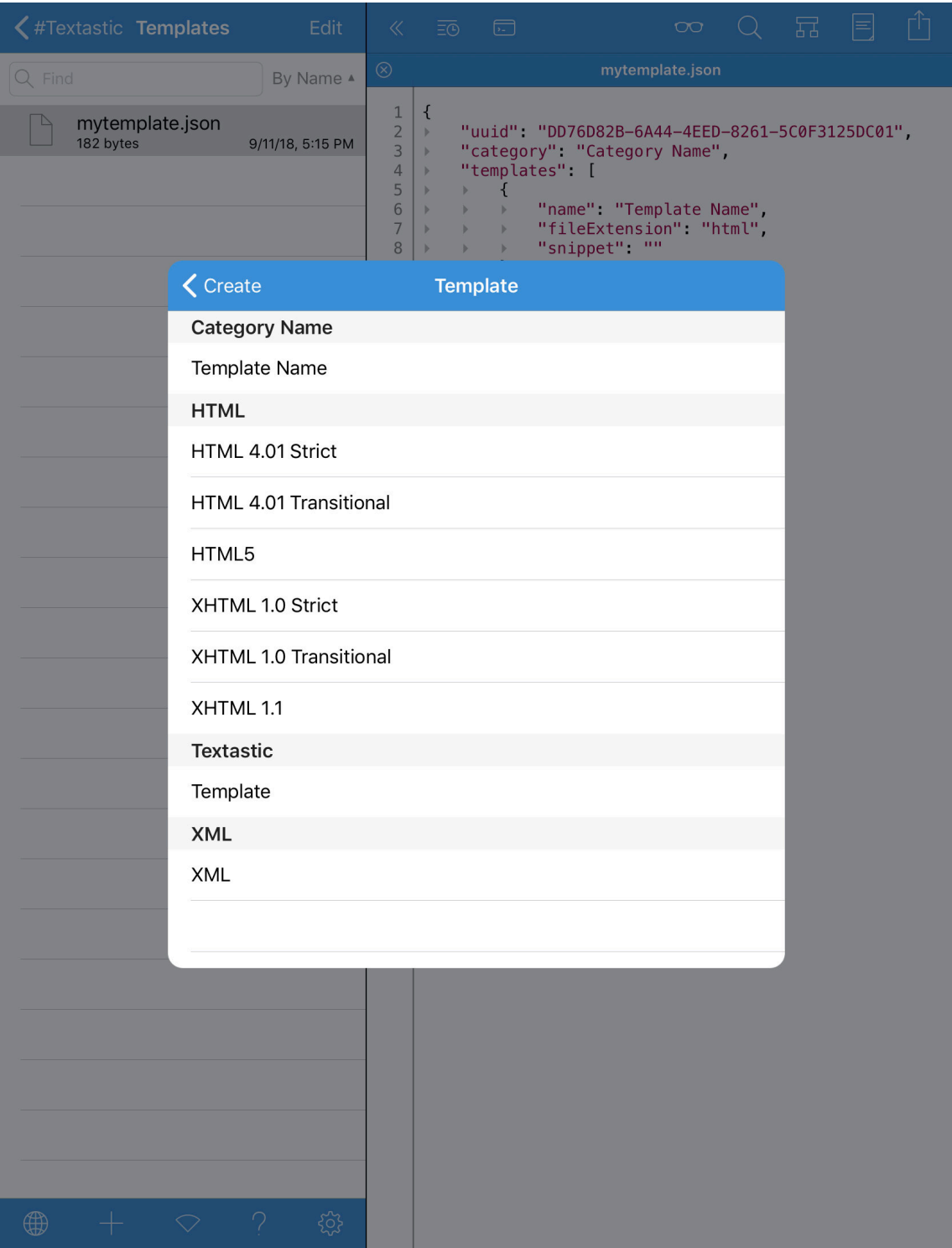
You can find a detailed description and example files at the Textastic-Customization GitHub page.

7.2 Installing Custom Fonts

You can choose your preferred font in the code editor settings and SSH terminal settings. In

addition to its built-in fonts, Textastic will also list all monospaced font installed on your device.

To install custom fonts system-wide on your iOS device you need to use *configuration profiles*.

Tip Fonts installed using configuration profiles are available to all apps on your device. For example, you can also use these fonts in Apple's Pages or Microsoft Word.

Use apps like **Fontcase** or **AnyFont** on your iPad or iPhone, or **Apple Configurator 2** on your Mac to install custom fonts and they will be available in Textastic.

7.2.1 Using Fontcase

Probably the easiest way to install a custom font using a configuration profile directly on your iOS device is using the free and open source **Fontcase** app.

Download Fontcase from the App Store.

Launch **Fontcase** and you are presented with this welcome screen that you confirm with *OK*:
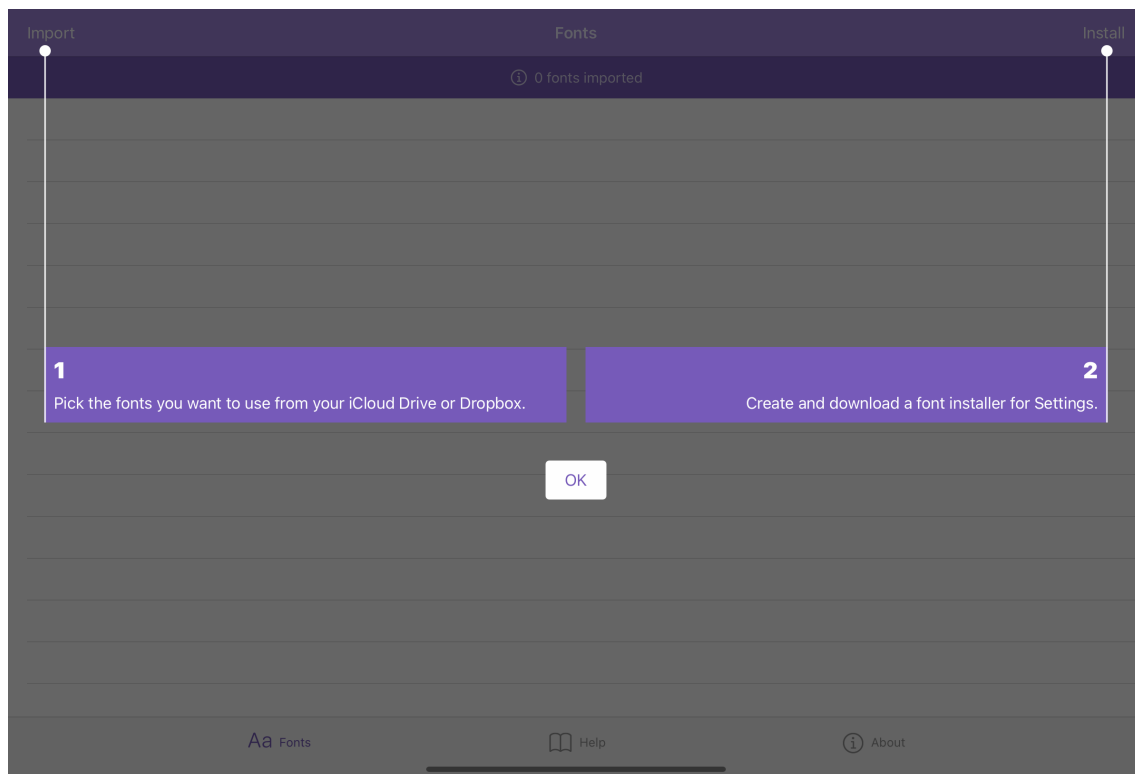
Tap the *Import* button in the top left corner and select the font files you want to install.

In this example I selected the regular, bold, italic, and bold italic OpenType (*.otf*) font files of the IBM Plex Mono font from a folder in iCloud Drive. Tap *Open*.

Import   Install

< Browse   Deselect All   **4 Items Selected**   **Open Done**

🔍 Search   🎤

✓ **Ag** IBMPlexMono-Bold.otf   ☁   9/28/21, 1:47 AM
              73 KB

✓ **Ag** IBMPlexMono-BoldItalic.otf   ☁   9/28/21, 1:47 AM
              79 KB

✓ *Ag* IBMPlexMono-Italic.otf   ☁   9/28/21, 1:47 AM
              78 KB

✓ Ag IBMPlexMono-Regular.otf   9/28/21, 1:47 AM
              71 KB

     📄 license   9/28/21, 1:47 AM
              4 KB

                    5 items, 1.79 TB available on iCloud

       Aa Fonts   📖 Help   ⓘ About

You will see a list of the imported font files.
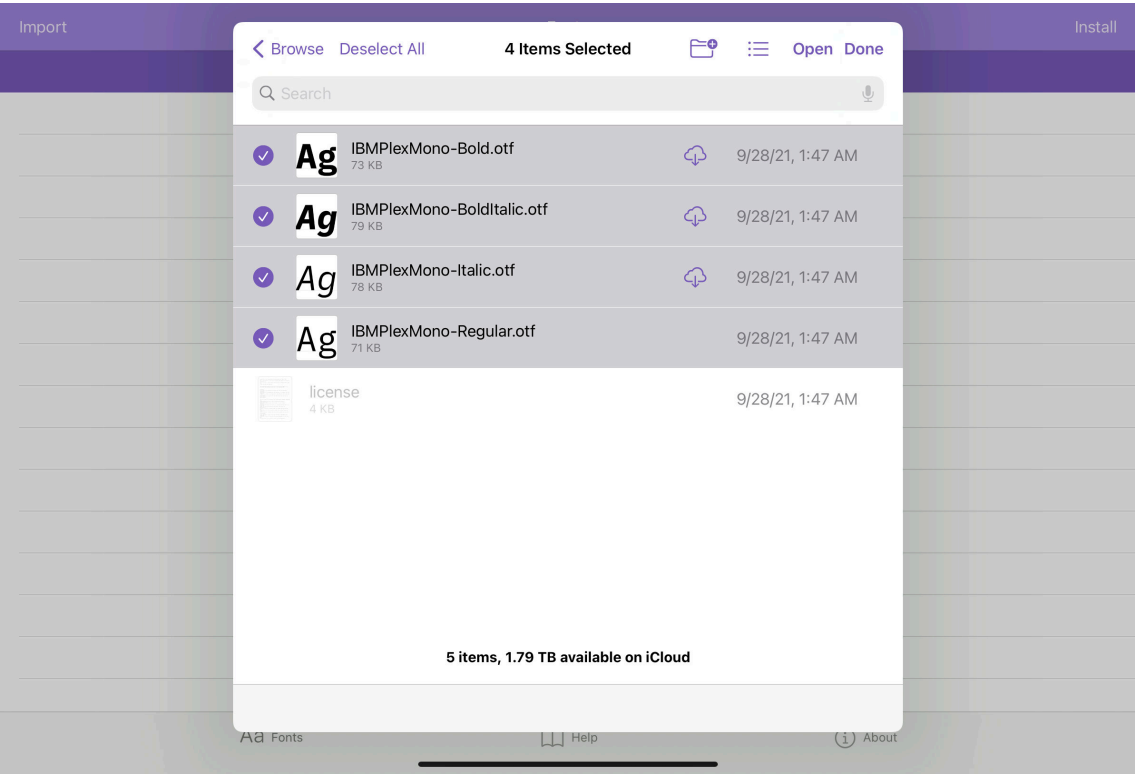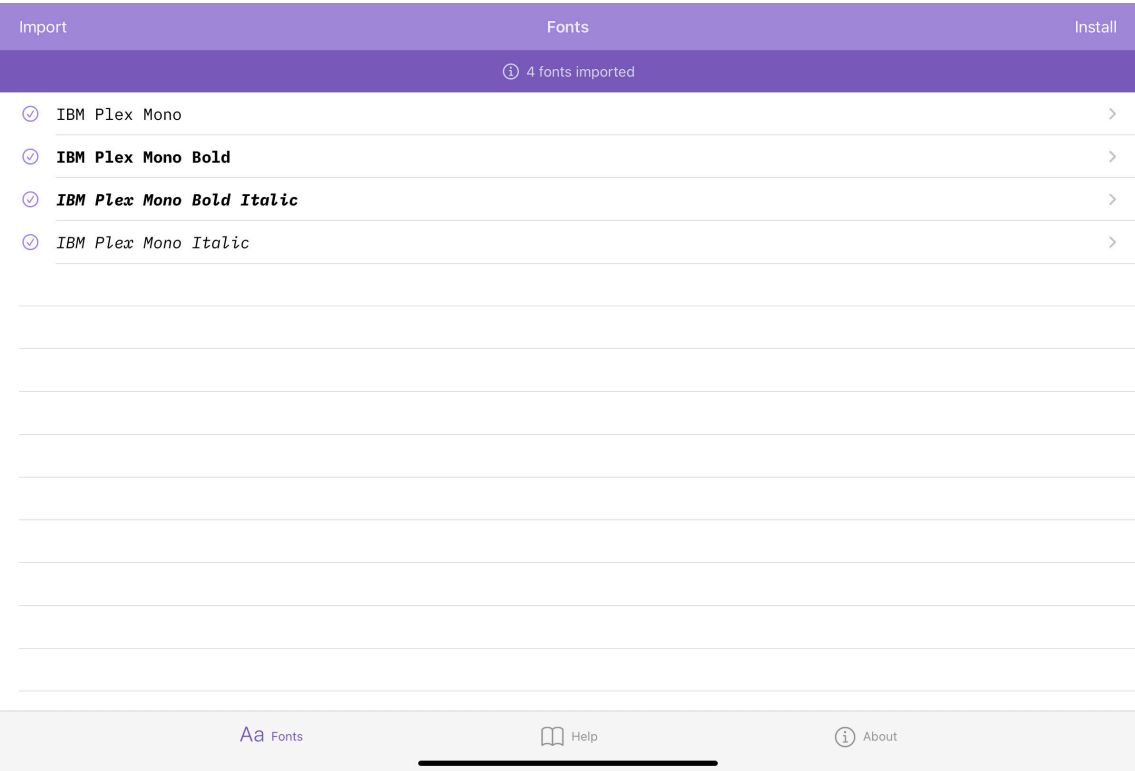
Import   Fonts   Install

                                 ⓘ 4 fonts imported

⊘ IBM Plex Mono   >

⊘ **IBM Plex Mono Bold**   >

⊘ ***IBM Plex Mono Bold Italic***   >

⊘ *IBM Plex Mono Italic*   >

       Aa Fonts   📖 Help   ⓘ About

Tap *Install* in the top right corner. Fontcase will package all imported font files in a configuration profile and show a local web page:
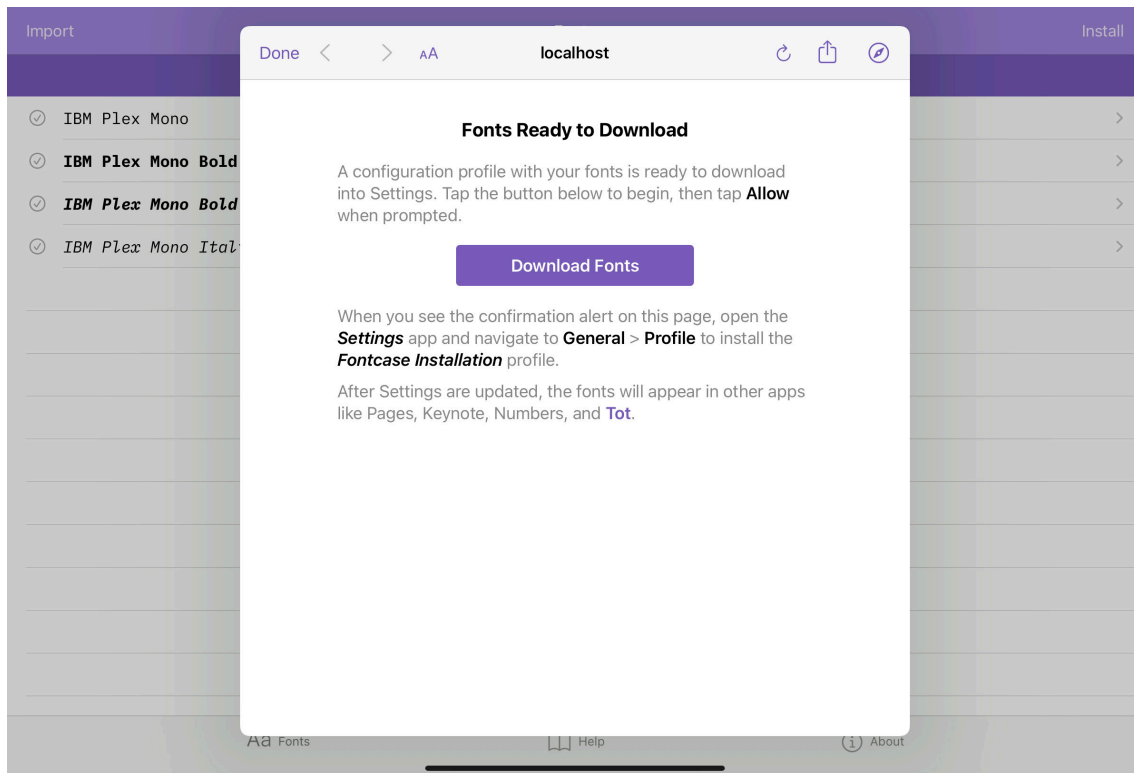
Tap *Download Fonts*. This will show a prompt that you need to confirm by tapping *Allow*.
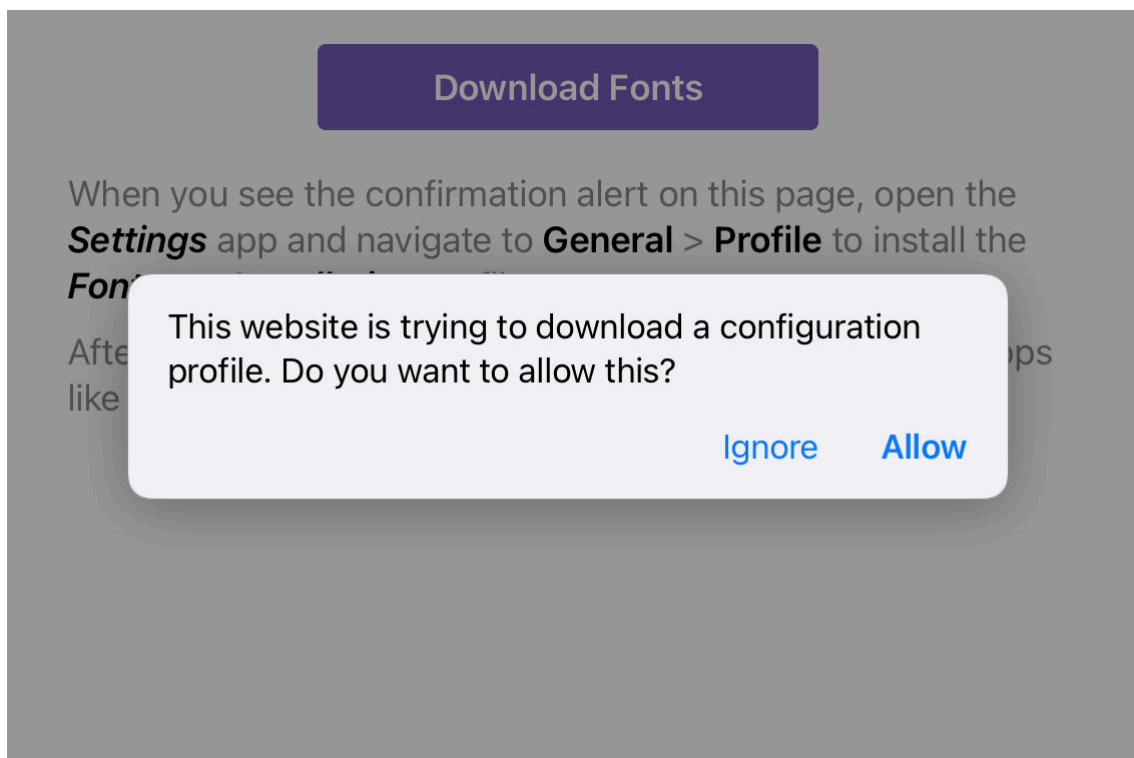
Switch to the `Settings` app. There should be a "Profile Downloaded" prompt that you can tap.

Tap *Install* and follow the instructions to install the profile. You may need to enter the passcode of your device in order to install the profile.

Cancel	Install Profile	Install

Fontcase Installation

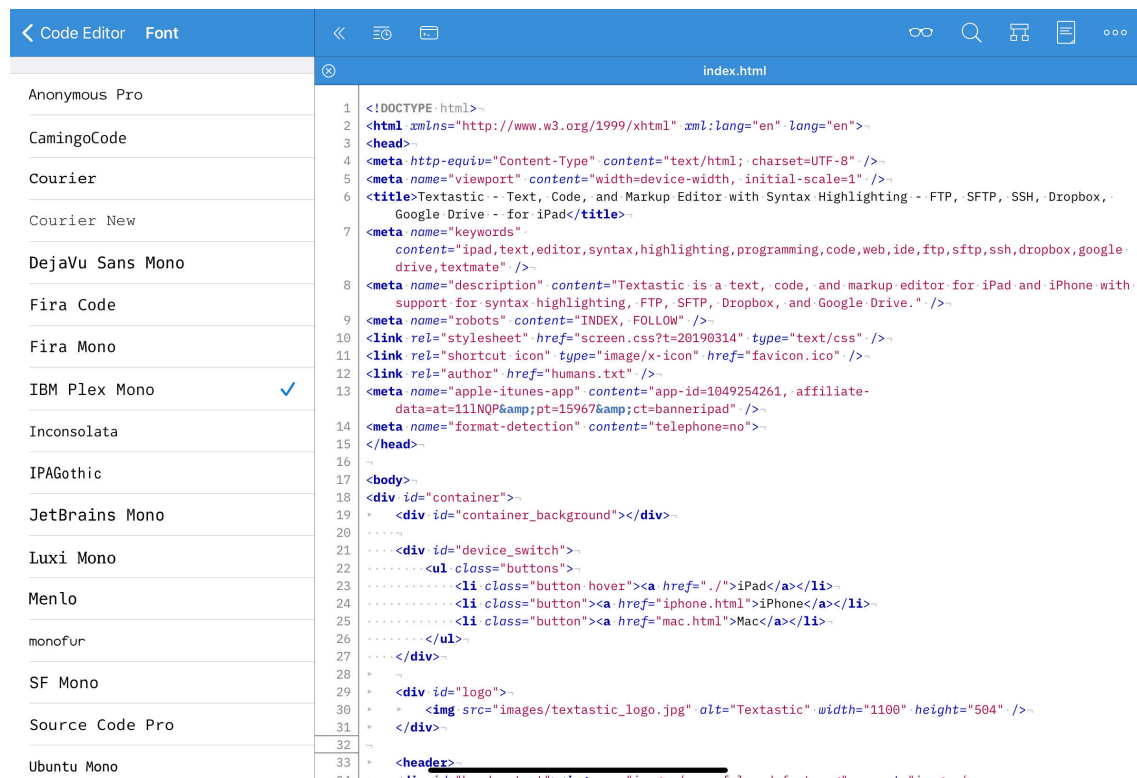| Signed by	**Not Signed**
Description	This profile installs the fonts managed by Fontcase.
Contains	4 Fonts

More Details ›

Remove Downloaded Profile
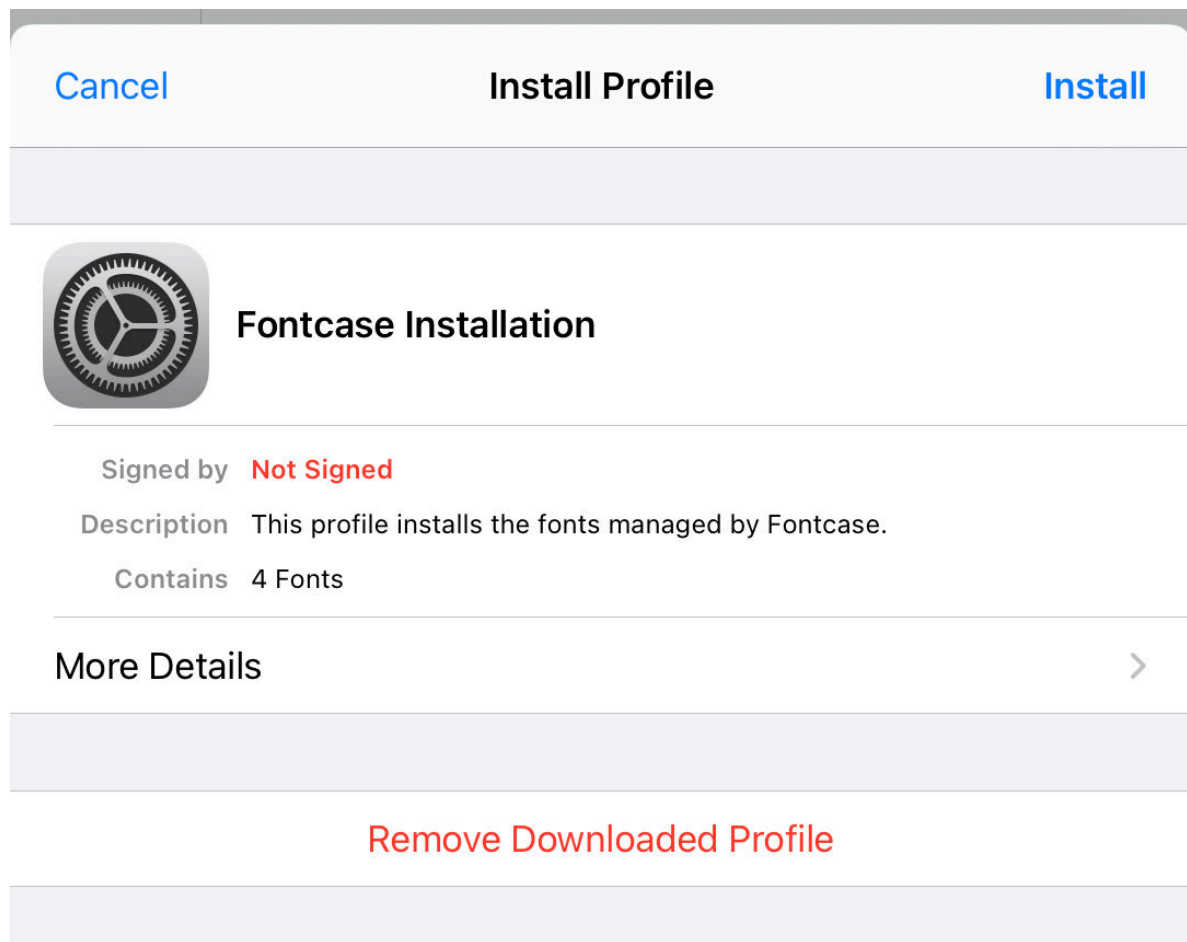
If you now launch **Textastic** and go to **Settings** ▸ **Code Editor** ▸ **Font**, the newly installed font will be available:

7.2.2 Using Apple Configurator 2

Install Apple Configurator 2

Download Apple Configurator 2 from the Mac App Store for free.

Create a Font Configuration Profile

In **Apple Configurator 2** on your Mac, choose **File ▸ New Profile**. A new configuration profile document window appears.

In the **General** settings pane, fill in the *Name* and *Identifier* fields.

Select **Fonts** from the list on the left, click *Configure*, then select one of the font files you want to add as a payload to the configuration profile.

Use the + button next to the font file to add additional font files.

In this example you can see the regular, bold, italic, and bold italic OpenType (*.otf*) font files of the IBM Plex Mono font added to the profile:
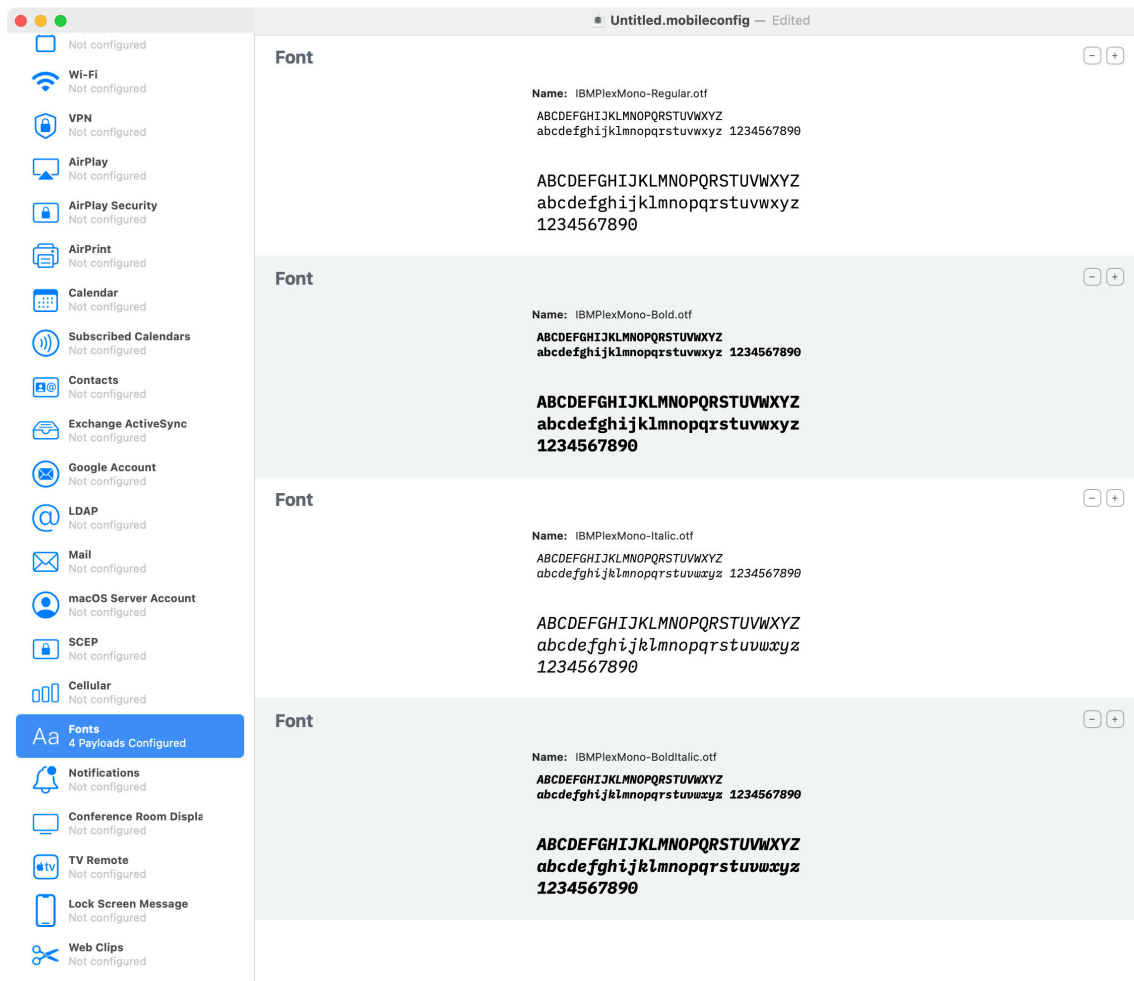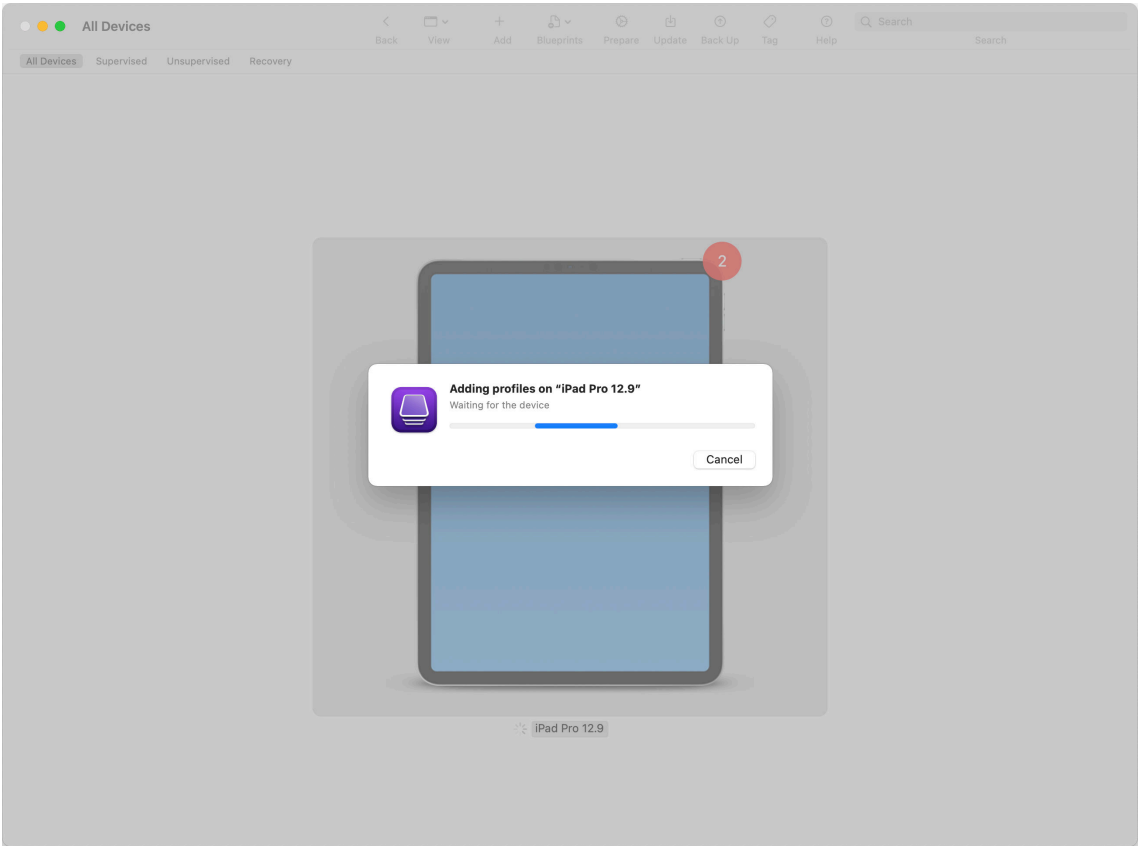
Choose **File** ‣ **Save**, name the profile, choose where to save it, then click Save.
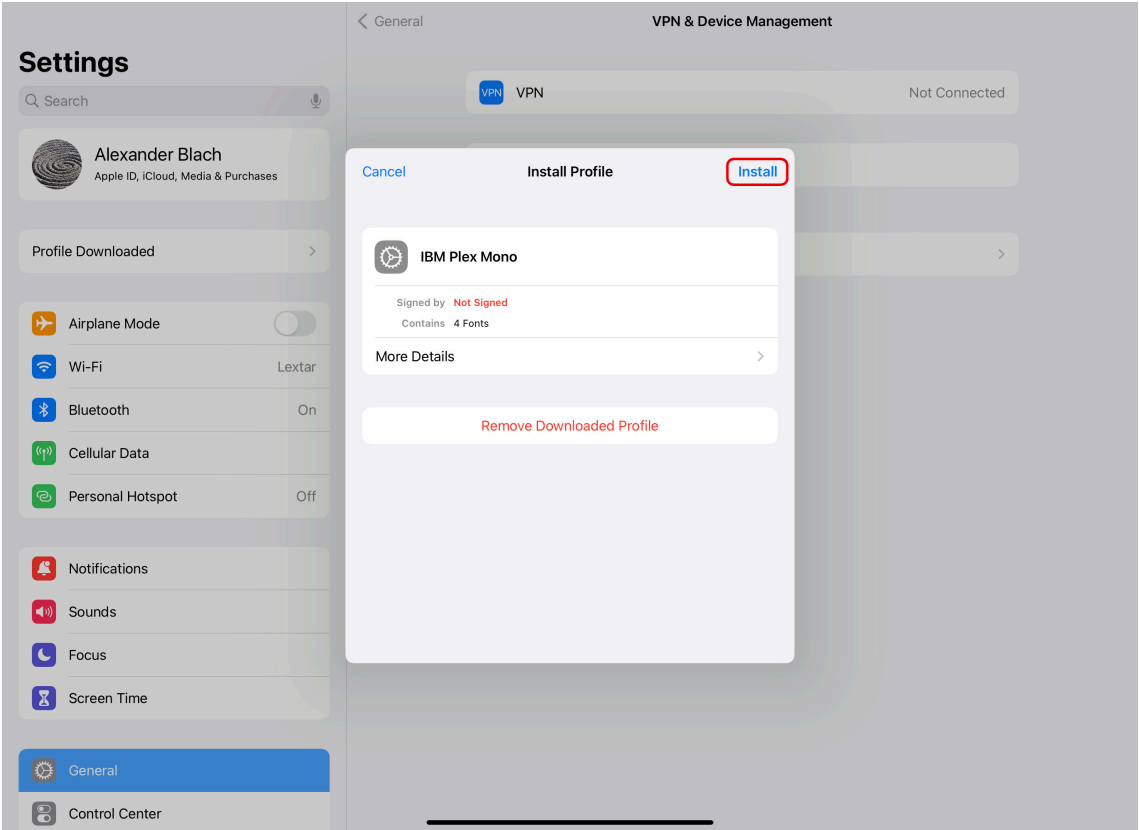
Install Font Profile

To install the font configuration profile created in the previous step, *connect your iPad or iPhone to your Mac*.

Select your device in Apple Configurator 2.

Choose **Actions** ‣ **Add** ‣ **Profiles**, select the previously saved profile, then click Add.

Confirm the installation of the profile in the **Settings** app on your device.

8 Connecting via Wi-Fi / built-in WebDAV server

The built-in WebDAV server allows you to easily transfer files to the Local Files location in Textastic from your Mac or PC via Wi-Fi.

macOS and Windows include WebDAV clients so you can use **Finder** and **Windows Explorer** to manage the files in Textastic.

Textastic is tested against macOS Finder, Windows Explorer, Cyberduck, BitKinex and the Total Commander WebDAV plugin.

Textastic also includes a WebDAV client in addition to the WebDAV server.

8.1 Starting the WebDAV Server

Tap the *WebDAV Server* button in the bottom toolbar to open the WebDAV screen.

You can also start the WebDAV server by going to **Settings ▸ WebDAV server** (see WebDAV server settings).
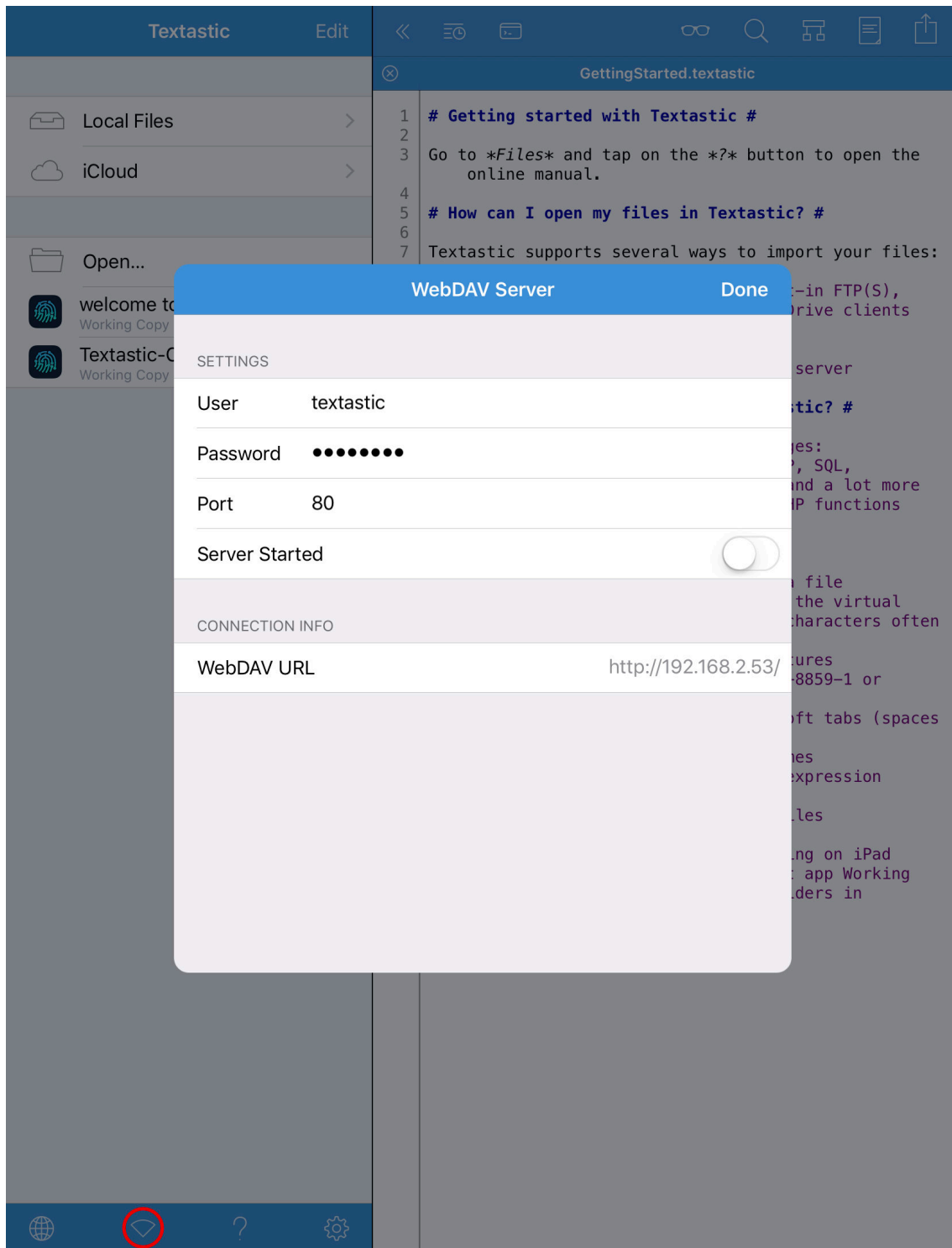
The WebDAV server always asks for user credentials. By default, the *User* is `textastic` and the *Password* is `password`. It's recommended to change these settings before starting the server.

You can choose a custom *Port* or use the default port 80.

Start the WebDAV server by flipping the *Server Started* switch to ON.
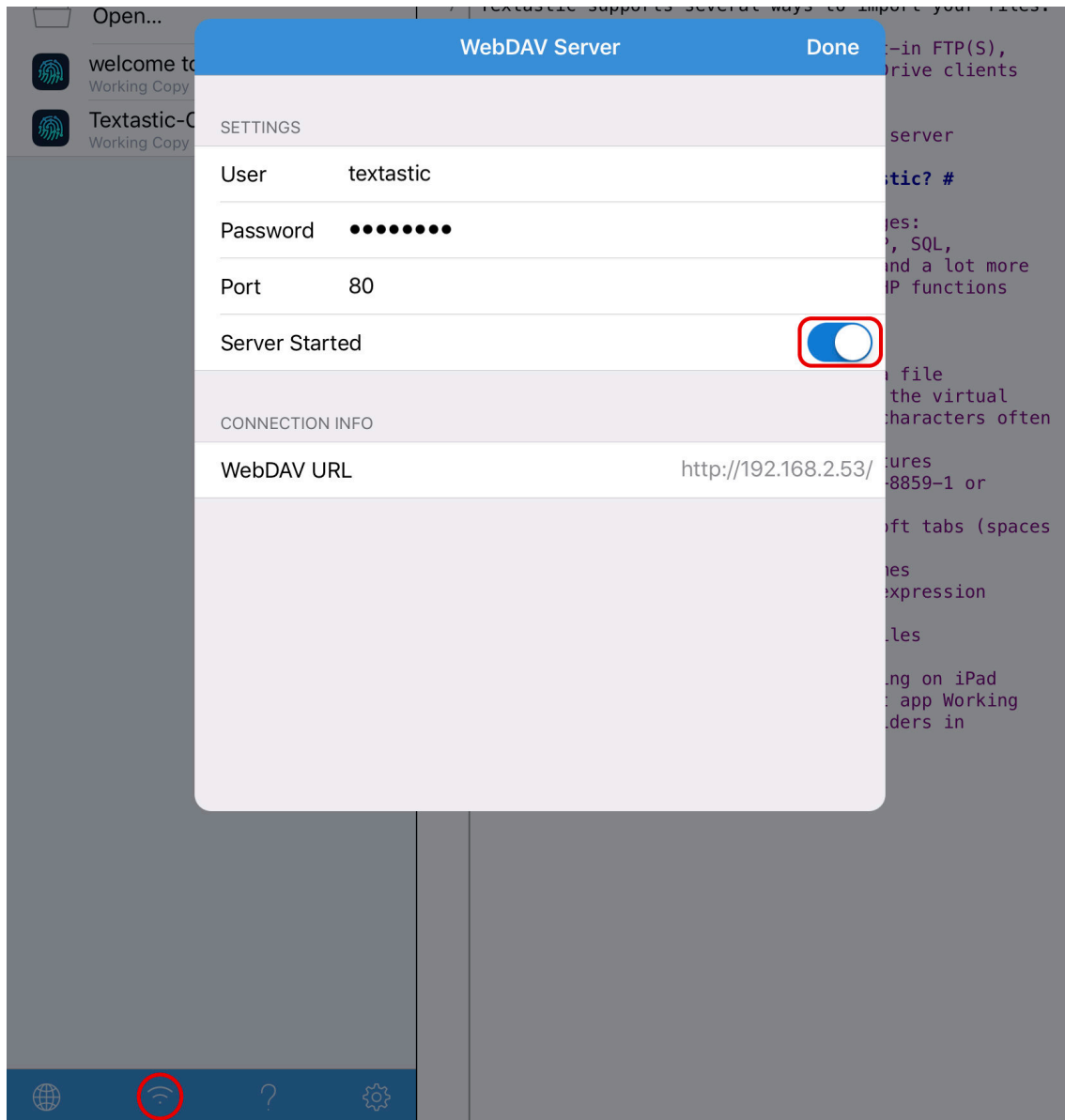
The button's icon in the bottom toolbar will change to indicated that the server is started.

Note Your iOS device will not go to sleep as long as the WebDAV server is running and Textastic is the foreground app. Use the power button on your iOS device to manually put it to sleep.

The WebDAV server is only running when Textastic is running in the foreground.
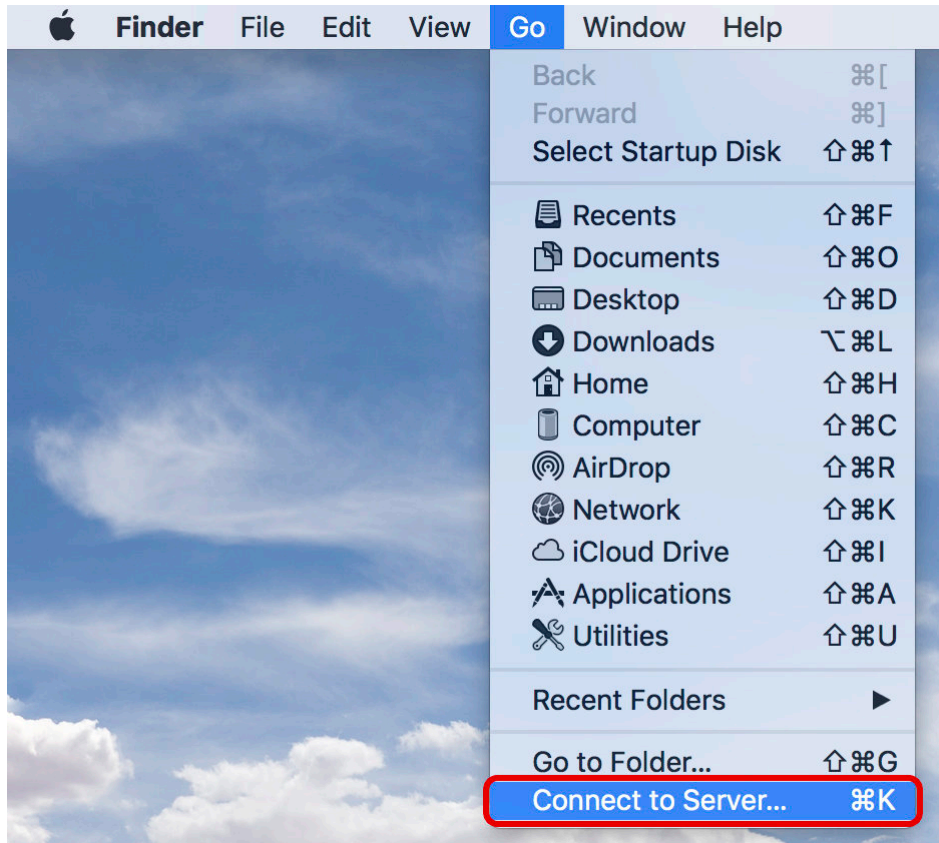
8.2 Connection Requirements

Your Mac or PC must be connected to the **same network** as your iOS device running Textastic.
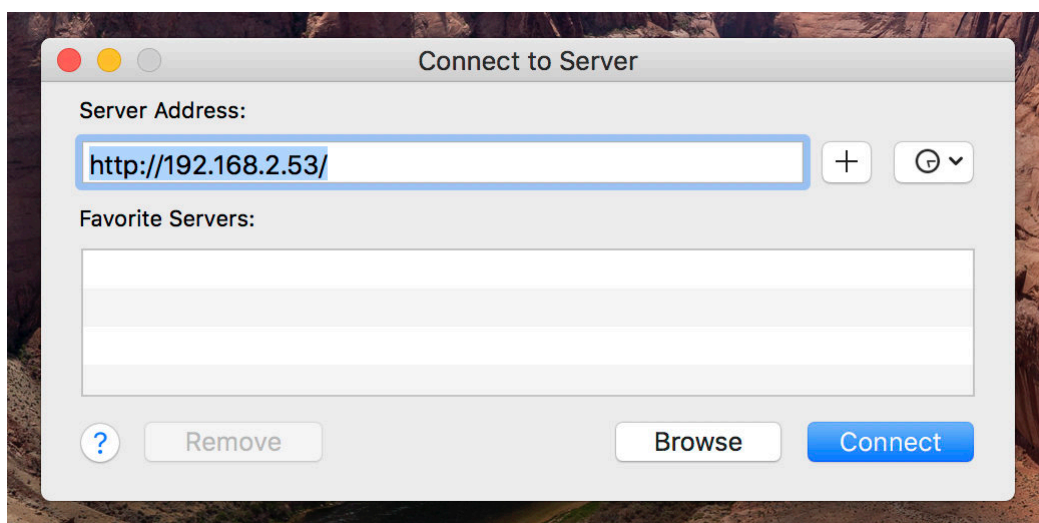
The iPad or iPhone device must be connected via **Wi-Fi**.

8.3 Connecting to Textastic with Finder

- Switch to Finder
- Choose **Go** ‣ **Connect to Server...**

- Enter the URL from the Textastic WebDAV screen in the *Server Address* field
- Click *Connect*

- Finder may warn you that this is an unsecured connection because the WebDAV server does not use SSL. Click *Connect*

- Enter the *Name* and *Password* you chose when setting up the WebDAV server (defaults are `textastic` and `password`).

- Click *Connect*

You can now use Finder to manage the files.

8.4 Connecting to Textastic with Windows Explorer

Windows users have two options for connecting to the Textastic WebDAV server: You can either map a network drive, or create a network location. Both methods are described below.

This should work in Windows Vista, Windows 7, Windows 8, and Windows 10.

8.4.1 Method A: Map a Network Drive

- Open Explorer and go to *This PC* or *Computer*.

- Click on the *Map Network Drive* button in the toolbar.

- Enter the Textastic WebDAV URL into the *Folder* field.

- Click *Connect using a different username* to enter your username and password.

- Click *Finish* to create the new drive letter.

This will map Files to a network drive with a name such as: `DAVWWWRoot` `(\\192.168.2.53) Y:`

To reconnect in the future just open this network drive from Computer.

8.4.2 Method B: Create a Network Location

- Open Explorer and go to *This PC* or *Computer*.

- Click on the *Map Network Drive* toolbar item.

- In the Map Network Drive window, click *Connect to a Web site that you can use to store your documents and pictures*, click *Next*.

- Select *Choose a custom network location*, click *Next*.

- In the Internet or network address field, enter the Textastic WebDAV URL

- You will be prompted for your username and password.

- Enter a name for the connection and click *Next*.

- Ensure that *Open this network location when I click Finish* is checked and click Finish.

In the future you can connect simply by opening "My Network Places" and double-clicking the network location you created. You may need to select "Refresh" from the window's context-menu to ensure that Windows has an up-to-date view of your files and folders.

Note If you change your username or password you may need to re-create the network drive or network location to gain access to the files.

9 Getting Help

If your questions are not answered in the manual or you want to report a bug, here are some additional resources.

9.1 Feedback Forum

There is a searchable public support forum at feedback.textasticapp.com where many of your questions might already be answered.
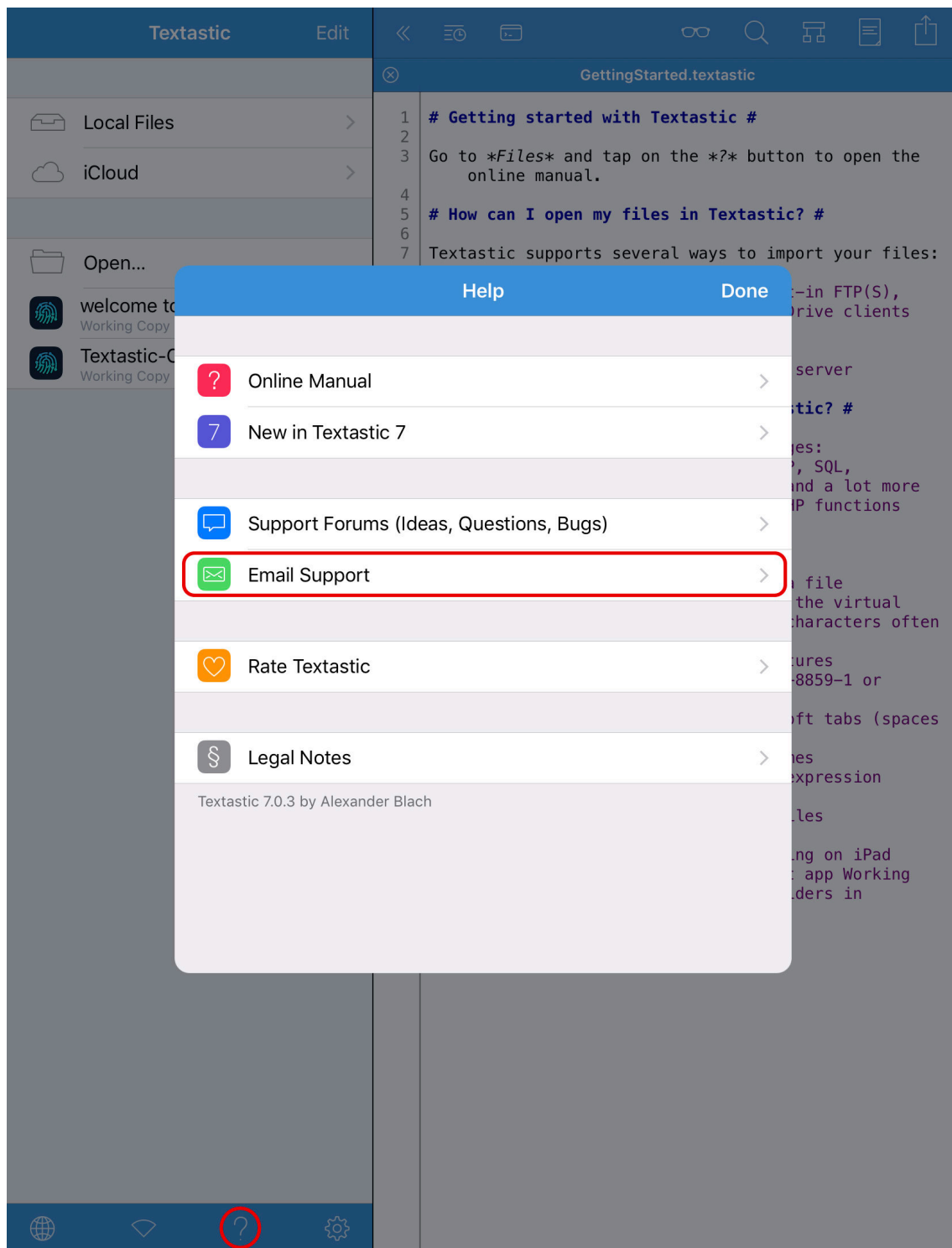
Use it to send me **ideas**, **questions** or **bugs** - either publicly or in private.

Also make sure to vote for the features you want to see implemented in future versions of Textastic.

9.2 Email Support

If you prefer, you can also send me an email:

- In Textastic, tap the *?* button in the bottom toolbar
- Tap *Email Support*

Or just send an email to support@textasticapp.com.

Important If you need help, please send me as much information as possible: detailed device information, step-by-step instructions, screenshots, and even small videos always help.