

# LocalizeTranslator

1.0

Documentation

Thank you for purchasing LocalizeTranslator from the Mac App Store!

This instruction manual can be accessed from the Help menu.

LocalizeTranslator uses the Google Cloud API for Translation.

\*charges to your google account may apply

Here is what you need to get started:

## 1. Create a google cloud account.

- An account can be created here: <https://cloud.google.com/>
- Once the account is created, go to the console.
- Create an app if one does not exist.
- Go to the APIs & Services Dashboard (click the menu to find this option)
- Click Enable APIs and Services
- Search for Cloud Translation API, and add it to your APIs and Services.

## 2. Enable the Cloud Translation API

- Make sure to enter your billing information if you have many translations
- Generate the access key to the API in a JSON file. Instructions from Google: <https://cloud.google.com/docs/authentication/getting-started>
- Save the key somewhere on your computer. You will select this file later in LocalizeTranslator.

### 3. Install gcloud on your Mac

- Make sure you have Python installed on your Mac
- Here is a link for installing gcloud: <https://cloud.google.com/sdk/docs/quickstart-macos>
- When gcloud finishes installation, make sure you take note of where the executable binary named “gcloud” is within the directory it was installed from. Check the bin folder.

### 4. Open LocalizeTranslator & Generate Token

- Every time this app is opened anew (or every ~12hr), you will need to generate a new token
- Google disables keys within ~12-24hr after they are generated
- Open Terminal. Do “`gcloud auth login`”
- After logging into gcloud with your Google account, make sure the correct project is set by id. It should tell you in the response in Terminal
- If you need to set a different project id, do “`gcloud config set project PROJECT_ID`”
- Remember that JSON file you downloaded previously? Run this command, but put in the path to your JSON file. `export GOOGLE_APPLICATION_CREDENTIALS="/home/user/Downloads/[FILE_NAME].json"`
- Here is the important step: Do “`gcloud auth application-default print-access-token`”
- If everything worked properly, Terminal should spit out an access token, which looks like some random text string. Copy and paste this into the Google Translate API Token field of LocalizeTranslator, and click the Set Token button.

## 5. Translate your Strings!

There are three different ways that you can use LocalizeTranslator.

### 1. Translate a Localizable.strings file

From the main window of LocalizeTranslator, you can take a Localizable.strings file that is in your language, and you can open it in LocalizeTranslator and convert it into any language. These files can then be dropped right into the Xcode lproj folders (appropriately named), or copied/pasted into the open files in Xcode.

### 2. Add New Strings to your existing localizations all at once in all languages

First, select the “Localize and Add Strings to Project” from the Tools menu. If you would like to add some new strings to your existing translations, all you need to do is select your Xcode project folder, and that folder should already contain some .lproj folders. Next, put in the language codes of the languages you would like to add the strings in. After that, copy and paste your strings into the strings to translate text view. Last, click the translate button, and watch your Xcode project magically gain all the new strings in every language.

### 3. Localize App Store Metadata

Select the “Localize Metadata” option from the Tools menu. From here, you can enter in all of the metadata from your app. Once it is entered, you can click “Save Metadata” so it is there on next app use. There are two main ways to localize your metadata: copying and pasting manually into App Store Connect, or you can use the Fastlane option. If you do not know what Fastlane is, please check out this link: <https://fastlane.tools>

With both methods, just type in or use the language codes in the language codes text view.

(Easy)

If not using fastlane, click the Translate button to translate into the first language selected, then copy your strings into App Store Connect. Click the next button, then translate for each language you would like to translate for.

(Advanced - Faster)

If using fastlane, open the “Localize and Add Strings to Project” from the Tools menu. Next, select your project path. This project path should have a fastlane folder within it. From here, you can go back to the “Localize Metadata” window and click the translate button. This will translate the metadata into all of the selected languages and store them in fastlane-supported txt files for automatic upload to the app store.

Make yourself familiar with the fastlane CLI tool and learn how to use the fastlane deliver command.

## Language Codes

### General Use

When using Localize Translator, all language codes that are entered as destination languages for translation should be in the appropriate format for the Google Cloud Translation API.

### Metadata Use

Fastlane uses the App Store Connect language codes, which are different from the Xcode language codes and the Google Cloud Language Codes. Here is a list of the available language codes, and the differences between them. Even though you can translate your metadata into any language that Google supports, the App Store only supports localized metadata in a limited selection of languages. Check figure 1A on the next page for more details.

Currently, you will need to manually go through your metadata language-code folders and make sure they are properly formatted for App Store Connect and that there are no unavailable languages in there. Without these common mistakes solved, fastlane deliver may arise some errors.

The Language Codes in three different formats will be provided below on Figure 1B. These will come in handy.

### Figure 1A

- English
- Australian English
- Brazilian Portuguese

- Canadian English
- Canadian French
- Danish
- Dutch
- Finnish
- French
- German
- Greek
- Indonesian
- Italian
- Japanese
- Korean
- Malay
- Mexican Spanish
- Norwegian
- Portuguese
- Russian
- Simplified Chinese
- Spanish
- Swedish
- Thai
- Traditional Chinese
- Turkish
- UK English (British English)
- Vietnamese

# Figure 1B

Language	Xcode .lproj	App Store Connect (fastlane)	Google API
Chinese (Simplified)	zh-Hans	zh-Hans	zh
Chinese (Traditional)	zh-Hant	zh-Hant	zh-TW
Danish	da	da	da
Dutch	nl	nl-NL	nl
English (Australia)	en-AU	en-AU	en
English (Canada)	en-CA	en-CA	en
English (U.S)	en	en-US	en
English (U.K.)	en-GB	en-GB	en
Finnish	fi	fi	fi
French	fr	fr-FR	fr
French (Canada)	fr-CA	fr-CA	fr
German	de	de-DE	de
Greek	el	el	el
Indonesian	id	id	id
Italian	it	it	it
Japanese	ja	ja	ja
Korean	ko	ko	ko
Malay	ms	ms	ms
Norwegian (Bokmal)	nb	no	no
Portuguese (Brazil)	pt-BR	pt-BR	pt
Portuguese (Portugal)	pt	pt-PT	pt
Russian	ru	ru	ru

Spanish (Mexico)	es-MX	es-MX	es
Spanish (Spain)	es	es-ES	es
Swedish	sv	sv	sv
Thai	th	th	th
Turkish	tr	tr	tr
Vietnamese	vi	vi	vi



## Conclusion

Good luck translating all of your strings, so the world can see your software!

Support Email: [chrisgalz@gmail.com](mailto:chrisgalz@gmail.com)

Please contact support before leaving a review if you have an issue with the app. I'm always looking to improve the software.

\*charges to your Google account may apply for translations based on number of requests

\$20 per 1,000,000 characters is 2018 rate

<https://cloud.google.com/translate/pricing>

©2018 Playr Inc.