

Keyman Developer Tutorial
Packaging the Keyboard
Session 8

This session we will show how to build/package a keyboard for the Dagbani language of Ghana. We will do this by putting together a package for distribution.

What is a package?

The purpose of a package is to make installation of a keyboard, fonts, documentation, and on-screen keyboard as straightforward as possible for the end user. We need to keep this goal in mind as we work on all the aspects of a package.

A good package includes the following:

- The keyboard itself, both the desktop and mobile
- The readme.htm file (the file is visible before installation).
- The welcome.htm file (the file is displayed after installation). Any images or other files referenced by the welcome.htm file should be included as well.
- The On-Screen keyboard if relevant.
- The splash image, if relevant.
- Other files documenting the use of the keyboard, usually in the welcome.htm file.
- Font files to be used with the keyboard, if necessary

The keyboard files are the KMX compiled keyman desktop keyboard and the JS compiled Keyman touch layout keyboard. Both should be added to the package. Note that the keyboard icon file is included in the KMX file.

The readme.htm file is a short description of the package, its use restrictions and what it includes. Try to keep the readme under 10 lines long. The readme is displayed in the package install dialog and should be an html file for optimal formatting.

The welcome.html file is an introductory help file in our package. This file will be detected during installation and displayed (in a window roughly two thirds of the user's screen width) after the package installation completes successfully. Make sure that the design of the HTML file can be resized to fit the user's screen - avoid extra wide tables or wide fixed width elements.

Welcome.htm file will be accessible after installation from Keyman Configuration. The intention of welcome.htm is to provide instructions on getting started with our keyboards.

The On-screen keyboard (KVK file) associated with our keyboard should be added to the package.

The splash image is a 140x250 pixel image that is displayed when the package is installed, at the left of the Package Install dialog. Including a splash image makes our package look more professional and polished, so a splash image is recommended.

The two preferred documentation formats are HTML and PDF. We should avoid DOC, RTF, and other formats. Remember that HTML files can be displayed on any computer without additional software. PDF files require Adobe Reader or a compatible PDF viewer application. We may choose to include both. PDF documents often print better than HTML documents, but HTML documentation is more accessible and translates better to on-screen or web use.

If the special characters of the language we are supporting are not in fonts included with Windows, we need to add fonts that will display these characters. Installing fonts in Windows is tedious, so make sure that the users don't have to locate and install fonts themselves. TTF, OTF, and TTC fonts will be installed by the package installer and uninstalled when the package is uninstalled. A list of the fonts installed is displayed in the Install Package dialog.

Creating a package

We will construct a Keyman Package Source (KPS) file.

1. Start Keyman Developer.
2. In the Project menu, point to Recent Projects, click DagbaniTutorial.kpj.
3. In the Project - Keyboards dialog box, click Packaging.

The KMX file is a compiled keyboard file from the Keyman source file (KMN).

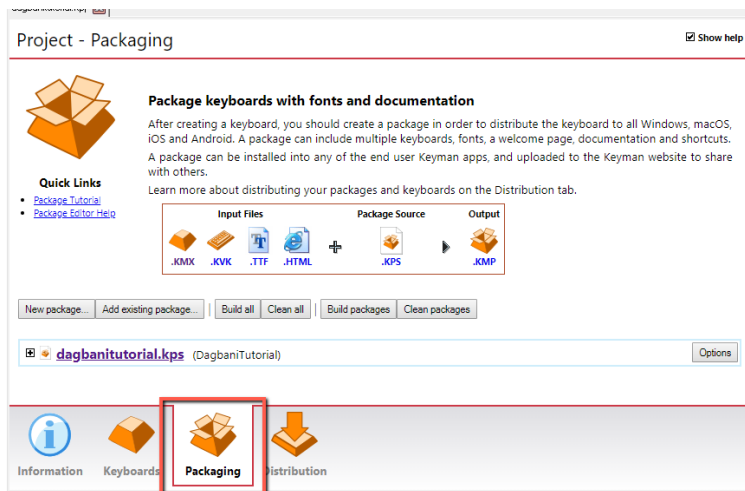
The KVK file is the visual keyboard file.

The TTF file is a TrueType font file.

The HTML files are documentation files.

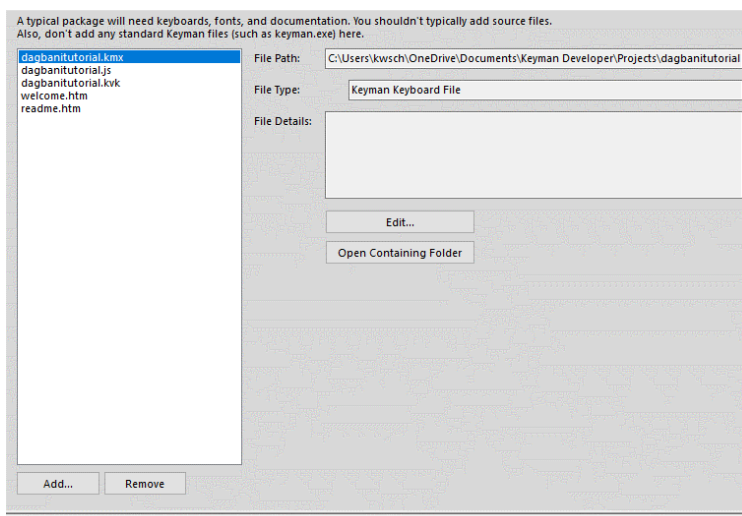
The KPS file contains the Keyman package source files.

The KMP file is a Keyman Package file for distributing keyboards in a zip format.

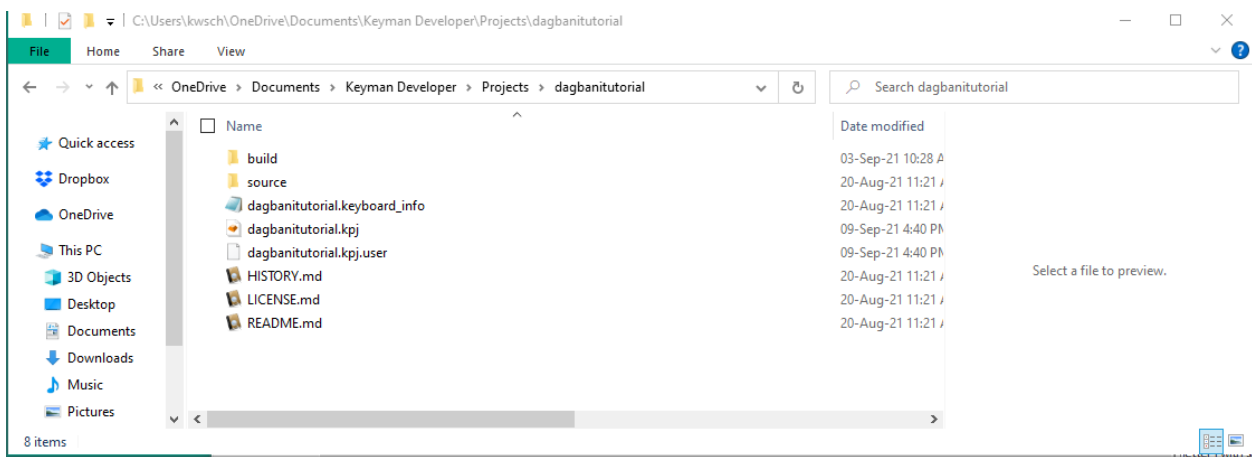


Building the Package File

1. We need to do some prep work before building the package. We will need to create several files. Click dagbanitutorial.kps. The Package Files pane appears. Then click Open Container Folder.



2. The Windows Explorer window appears. We are at the build folder. We want to go up one level to the dagbanitutorial folder.



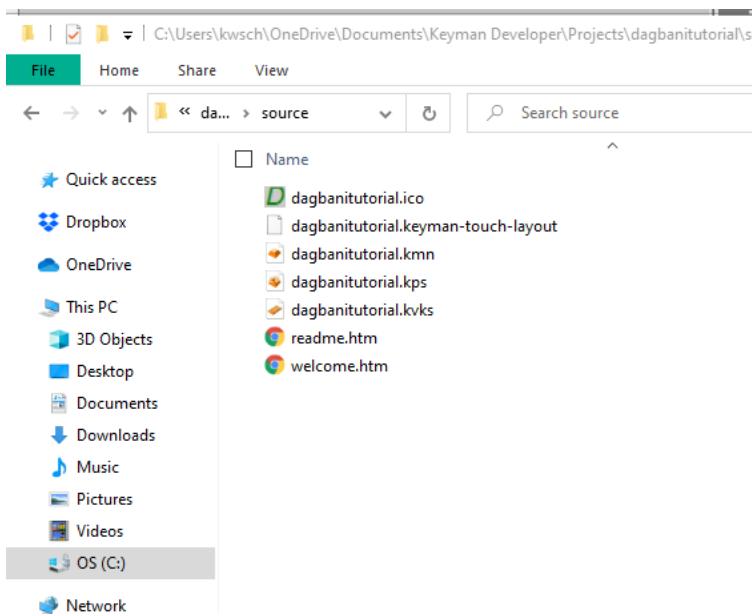
Open README.md in a text editor. We will probably want to update description section to describe what keyboard is about, saying something like DagbaniTutorial keyboard for the Dagbani language in Kenya based on US keyboard. When we do future versions of a keyboard, we will want to update the version number. Then save it and exit.

Open HISTORY.md in a text editor. Here is where we would keep a running history of all the changes to the keyboard. We will want to include the correct date. Make any needed changes. Then save it and exit.

Open LICENSE.md in a text editor. It will need copyright information update to include the correct date and the copyright holder. If we specify the copyright information when using the New Project wizard, it will take care of putting the copyright information where it needs to go. Make any other changes to the license information that is appropriate. Then save it and exit.

The three files above are internal documentation files that the user does not see.

3. Now we want to go to the source folder by double-clicking it.



Open readme.htm in a text editor. Update the description in line 18 to something like this DagbaniTutorial keyboard based on US keyboard. Make any other needed changes. Then save it and exit. This file is displayed when a user begins installation of the keyboard in a desktop environment

Open welcome.htm in a text editor. Update the description in line 18 to something like this DagbaniTutorial keyboard based on US keyboard. Copy and paste the text below under the

section where it says: Insert keyboard layout...

```
<img src='dagbanitutorialU_.png' alt='Keyboard Layout'>
<img src='dagbanitutorialU_S.png' alt='Keyboard Layout'>
```

This will show the user what the keyboard looks like. We are still in the editor. Now we want to show how to produce the special characters. We can do that with a link to a pdf file in the welcome.htm file. So, copy and paste the text below the description section.

```
<a href="file:dagbanitutorial.pdf">this document
for all the key combinations. </a>
```

Now copy the DagbaniTutorial.pdf to the source folder. Here is a [link to the file](#).

Make any other needed changes. Then save it and exit. We can check our work by opening the file in a browser to verify that it looks good and that any links work.

4. Now we need to create some help documentation that the Keyman repository will need. This is what we will be doing next.

Create a new folder within the source folder called help. Go to the help folder. Create a new text file and name it dagbanitutorial.php. This file would normally be the name of the language for the keyboard. Open this file in a text editor. Copy and paste the following text into the new document PHP document. This is what will be display on the Keyman web site when help is clicked.

```
<?php
$pageName = 'DagbaniTutorial Keyboard Help';
$pageTitle = $pageName;
// Header
require_once('header.php');

<p>
DagbaniTutorial keyboard based on US keyboard.
</p>

<h1>Keyboard Layout</h1>

<h2>Desktop Keyboard Layout</h2>
<div id='osk' data-states='default shift'>
</div>

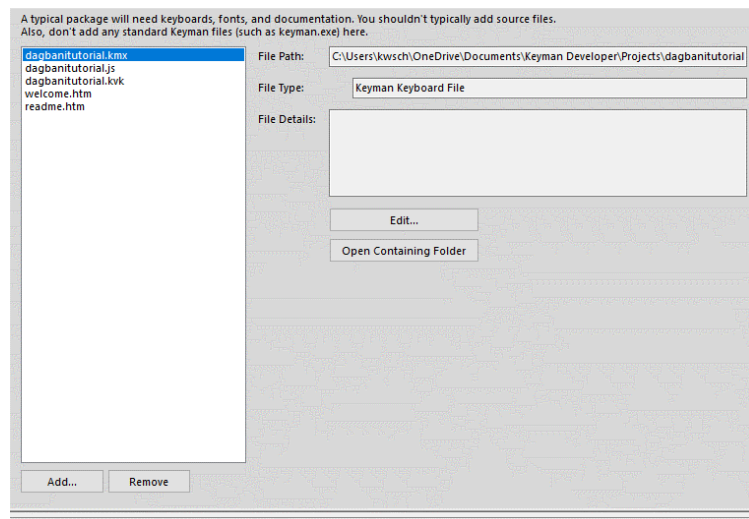
<h2>Mobile Keyboard Layout</h2>
<p>Due to the size and number of keys, some characters are hidden in the long press.
Press and hold on the key with a little dot on the top right to reveal them.</p>
<div id='osk-phone' data-states='default shift numeric'>
</div>
```

Modify as needed, particularly the correct data-states. They need to match the layers of the keyboard. If we needed to create a separate tablet and phone section in place of the mobile

section. We would need to change the Mobile Keyboard Layout to Tablet Keyboard Layout, adjusting by the data-states as needed. And then create a new section for the Phone Keyboard Layout, adjusting the data-states as needed. We would use osk-phone for phone layout and use osk-tablet for tablet layout.

Make any other needed changes. Then save it and exit. This file is for the users to see on the Keyman web site and to help them to know how to use this keyboard.

5. Now we are ready to create the package. In the Project - Keyboards dialog box, click Packaging. The Package Files pane appears. The welcome.htm and readme.htm are there because we used the New Project wizard. In order for the readme.htm to be displayed at install time, it needs to be selected on the Package Details page (Details tab of Packaging), under the Optional Information heading. Again this is done automatically if the New Project wizard was used.

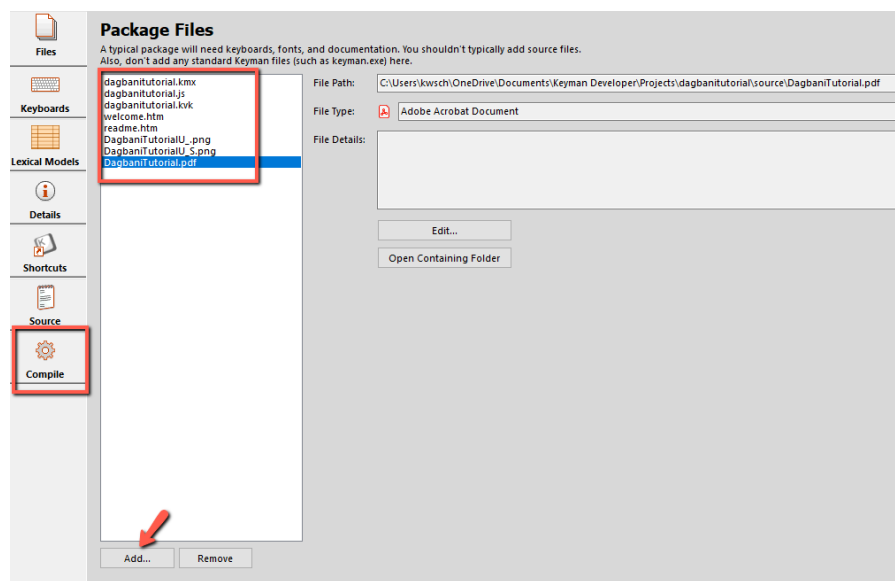


Here we have a list of files that will go into the package. Now we need to add other files to this list such as the PDF files, the font files, and the image files.

First, we will add the image files by clicking Add. The Add Files dialog box appears. Verify that we are at the source folder. Select dagbanitutorialU_.png and dagbanitutorialU_S.png. Then click Open.

Second, we will add the pdf file by clicking Add. The Add Files dialog box appears. Select DagbaniTutorial.pdf. Then click Open.

If we had any font files that need to be added, we would add them here.



Now that we have added the files we need to the packaging list, we want to save our work by clicking Save.

6. Now we are ready to compile the package by clicking Compile. The Compile Package pane appears. Then click Compile Package. Looking at the results, in the Message box, we verify that we have successfully compiled the package, noting the green line.

Then we see in the Target filename box, the name of KMP file. This is the file we can give to others for installing the keyboard on their computers. Once we have tested the keyboard package, we are ready to submit the keyboard to the Keyman keyboards repository so that it will be available to anyone with Internet access

Click Open build folder to locate the KMP file.