KDT15 Session 7

Keyman Developer Tutorial

Testing a Touch Keyboard

Session 7

This session we will test the touch keyboard for the Dagbani language of Ghana.

1. Start **Keyman Developer**.

2. In the **Project** menu, point to **Recent Projects**, click **DagbaniTutorial.kpj**.

3. In the **Project - Keyboard** dialog box, click **Keyboards**. Then click **dagbanitutorial.kmn**. The **Details** page appears.

4. Click **Build**. The **Build** page appears.

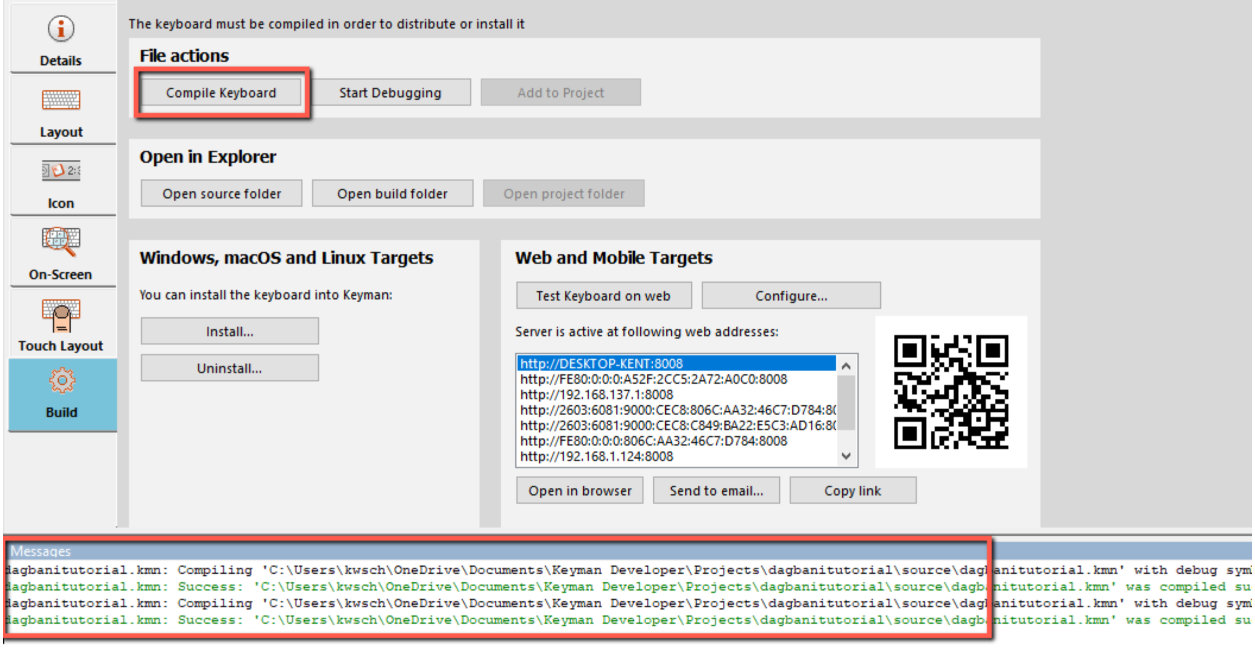
5. Click **Compile Keyboard**.

If the keyboard has compiled successfully, we will see two green messages saying *Success:* … in the **Message** pane. If the messages are red, the compilation fails. The error should indicate on what line in the code the error occurs. We will need to identify the error and then correct it and then recompile the keyboard.

We might see a warning like this: *Warning: line 0 warning 2092: Key "T_new_579" on layer "default", platform "phone", is a custom key but has no corresponding rule in the source*. This indicates that we forgot to enter the correct keycode for a new key we added. Double-check the long-press popup keys for one with a **T_new_NNN** code (T_new_579 in this example) and enter the desired **U_NNNN** code in the **Code** box.

The purpose of the T_new_NNN codes is so that we can write rules for these keys in the same way we do for physical keys. If we add these key references which we can customize to anything (like T_Hook_Y_UC) to the code, then we make rules for them alongside the physical keyboard rules in the **Code** tab.

We might also see a message like this: *Error: line 0 error 405A: Key "U+0181" on "phone", layer "default" has an invalid identifier.* This indicates that we attempted to change the keycode but mistyped the code. We may have gotten this more than once because we are used to typing **U+0181** to refer to a Unicode value, but the correct syntax for a keycode in Keyman Developer is **U_0181**.



The keyboard must be compiled in order to distribute or install it

You can install the keyboard into Keyman:

Server is active at following web addresses:

```
http://DESKTOP-KENT:8008
http://FE80:0:0:0:A52F:2CC5:2A72:A0C0:8008
http://192.168.137.1:8008
http://2603:6081:9000:CEC8:806C:AA32:46C7:D784:8(
http://2603:6081:9000:CEC8:C849:BA22:E5C3:AD16:8(
http://FE80:0:0:0:806C:AA32:46C7:D784:8008
http://192.168.1.124:8008
```

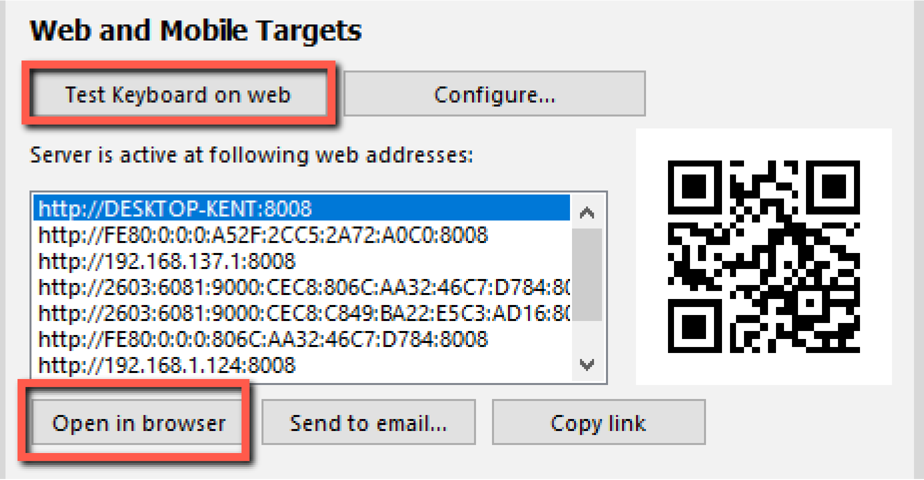
```
dagbanitutorial.kmn: Compiling 'C:\Users\kwsch\OneDrive\Documents\Keyman Developer\Projects\dagbanitutorial\source\dagbanitutorial.kmn' with debug sym
dagbanitutorial.kmn: Success: 'C:\Users\kwsch\OneDrive\Documents\Keyman Developer\Projects\dagbanitutorial\source\dagbanitutorial.kmn' was compiled su
dagbanitutorial.kmn: Compiling 'C:\Users\kwsch\OneDrive\Documents\Keyman Developer\Projects\dagbanitutorial\source\dagbanitutorial.kmn' with debug sym
dagbanitutorial.kmn: Success: 'C:\Users\kwsch\OneDrive\Documents\Keyman Developer\Projects\dagbanitutorial\source\dagbanitutorial.kmn' was compiled su
```

6. After a successful compilation, we are ready to test the touch keyboard. We need to follow these instructions very carefully.

In the **Keyboard** menu, the **Test Keyboard** option only tests the desktop versions of a keyboard, not the mobile versions. To test a mobile keyboard, we can use the Chrome browser on the Windows computer to simulate a mobile device.

If the default browser is not Chrome. This may or may not work. If not, we will need to copy the URL and paste it into Chrome.

In the **Web and Mobile Targets** box, click **Test Keyboard on web**. Then click **Open in browser**.

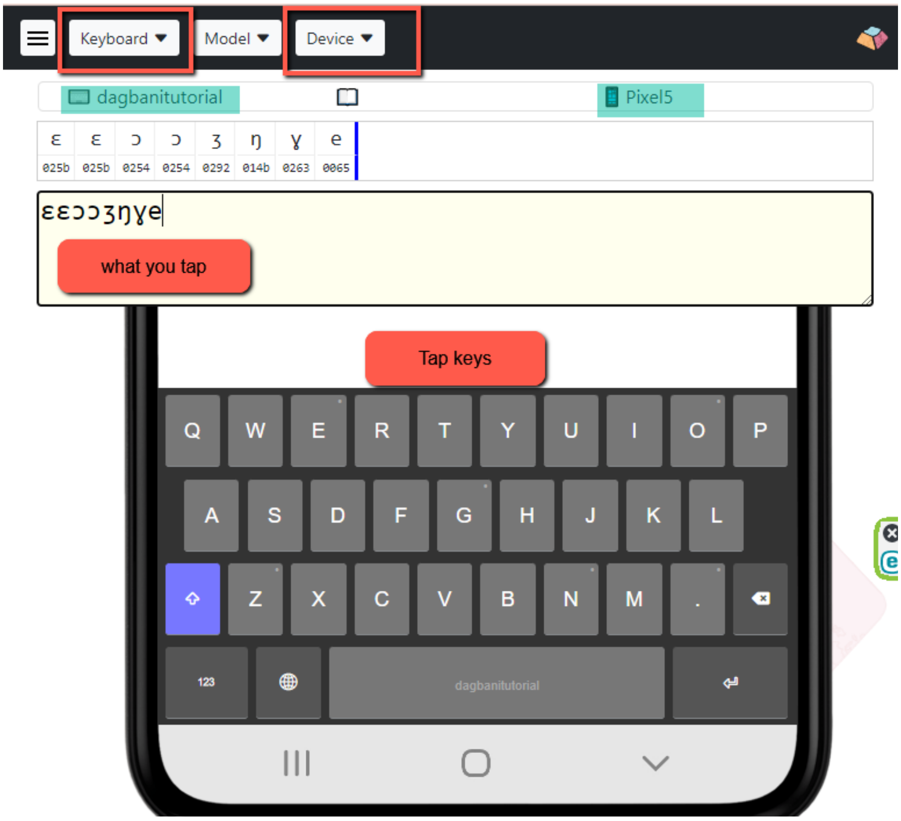


Server is active at following web addresses:

```
http://DESKTOP-KENT:8008
http://FE80:0:0:0:A52F:2CC5:2A72:A0C0:8008
http://192.168.137.1:8008
http://2603:6081:9000:CEC8:806C:AA32:46C7:D784:8(
http://2603:6081:9000:CEC8:C849:BA22:E5C3:AD16:8(
http://FE80:0:0:0:806C:AA32:46C7:D784:8008
http://192.168.1.124:8008
```

The browser appears. Now set the **Keyboard** to **dagbanitutorial**. Set the **Device** to **Pixel 5**. Now the mobile keyboard is ready for testing.

Then press various keys on the mobile keyboard to verify that the mobile keyboard is working as expected. Note that keys that have dot in the upper right-hand corner have long-press keys. If so, we are done testing. If not, we need to fix the problems in the Touch Layout section.



ɛ	ɛ	ɔ	ɔ	ʒ	ŋ	ɣ	e
025b	025b	0254	0254	0292	014b	0263	0065

ɛɛɔɔʒŋɣe

7. Close the browser.

We have completed our testing