1. The designer is open, and we have a screen with labels, buttons, etc.

Level of difficulty:   
Estimated time: 15 mn

[**www.codapps.io**](http://www.codapps.io)by Clement Levallois **@codapps\_io** [**www.coursera.org/learn/codapps**](http://www.coursera.org/learn/codapps)



**Module 7 - BONUS**

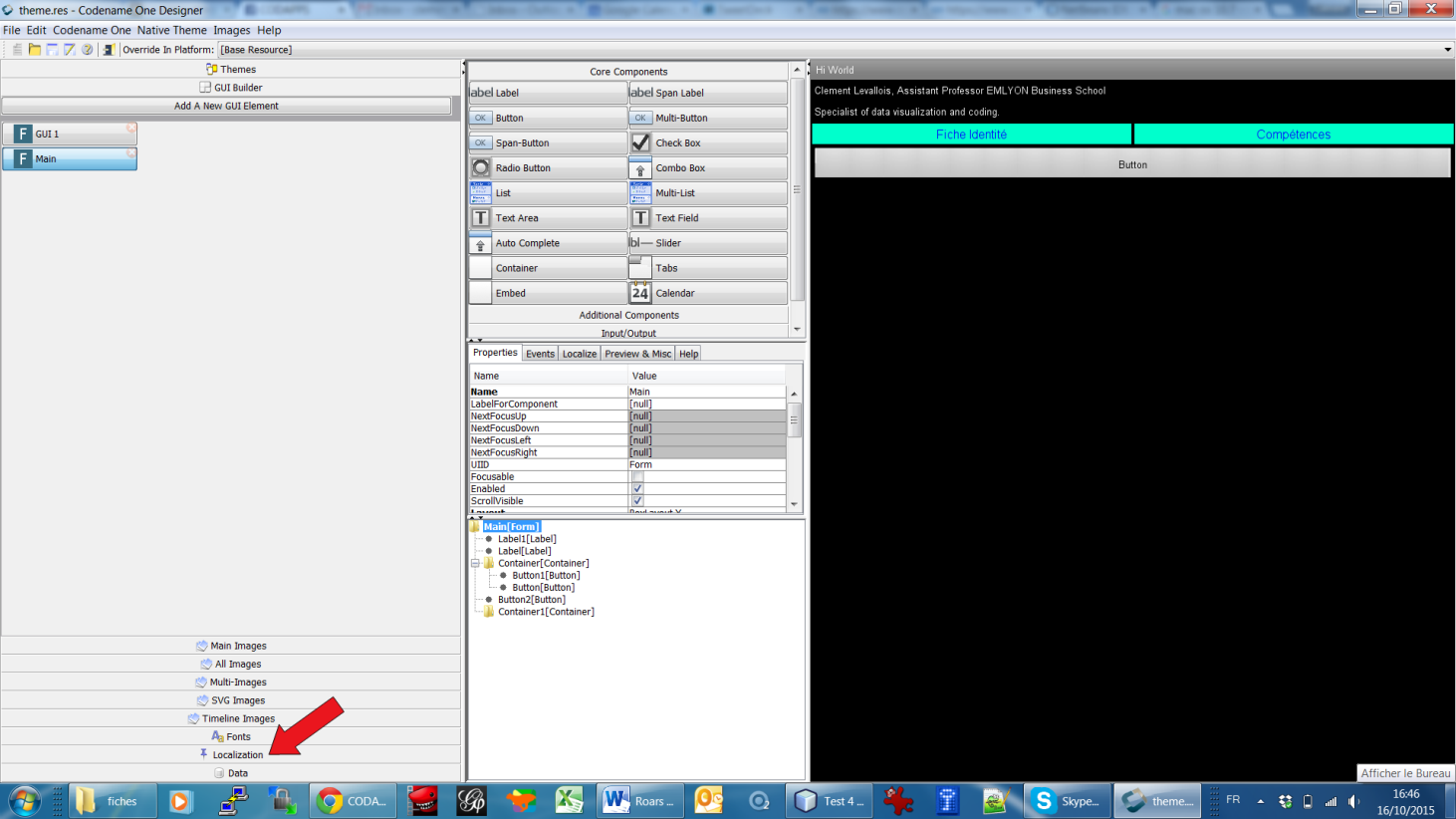
*How to make your app appear in different languages*

**We would like an app which appears in French on French phones, and in English on all phones which are not French phone. How to do that?**

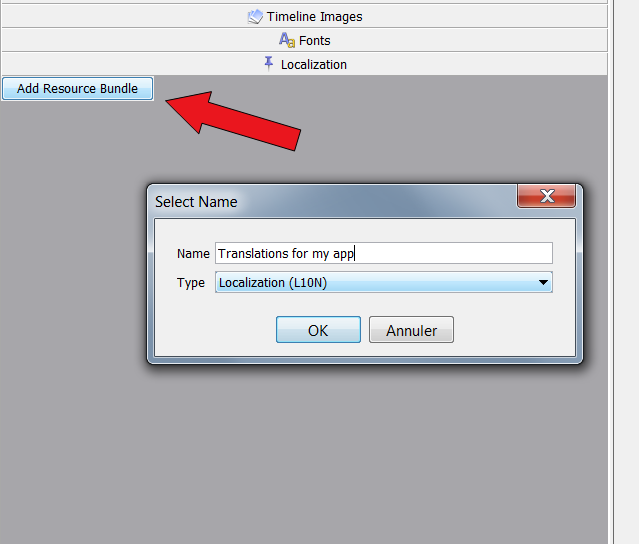
The general principle is to use the menu “**Localization**”:

* For each Label and Button on all screens, we translate the text in French and English
* We add a few lines of code in the file “StateMachine.java” in NetBeans to decide which language to use, depending on the language of the phone.

**Let’s do it. In the menu on the left, click on “Localization”:**



1. Then click on “**Add Resource Bundle**”, and give a name for your translations. Here I called them “Translations for my app”:



1. We click on “Synch UI”, this will show all elements of text in our app:

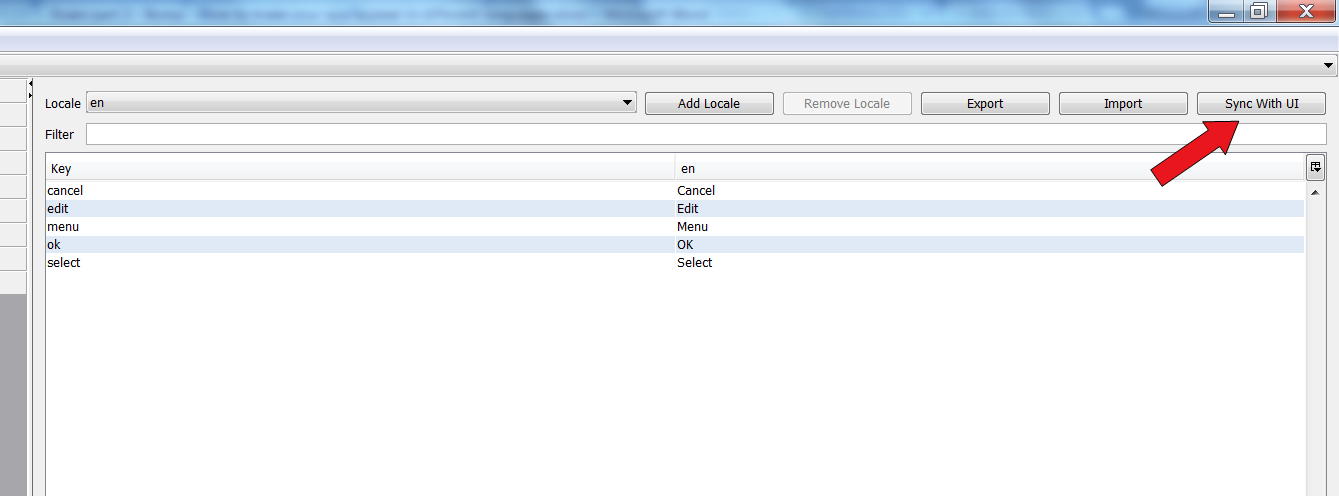
Level of difficulty:   
Estimated time: 15 mn

[**www.codapps.io**](http://www.codapps.io)by Clement Levallois **@codapps\_io** [**www.coursera.org/learn/codapps**](http://www.coursera.org/learn/codapps)



**Module 7 - BONUS**

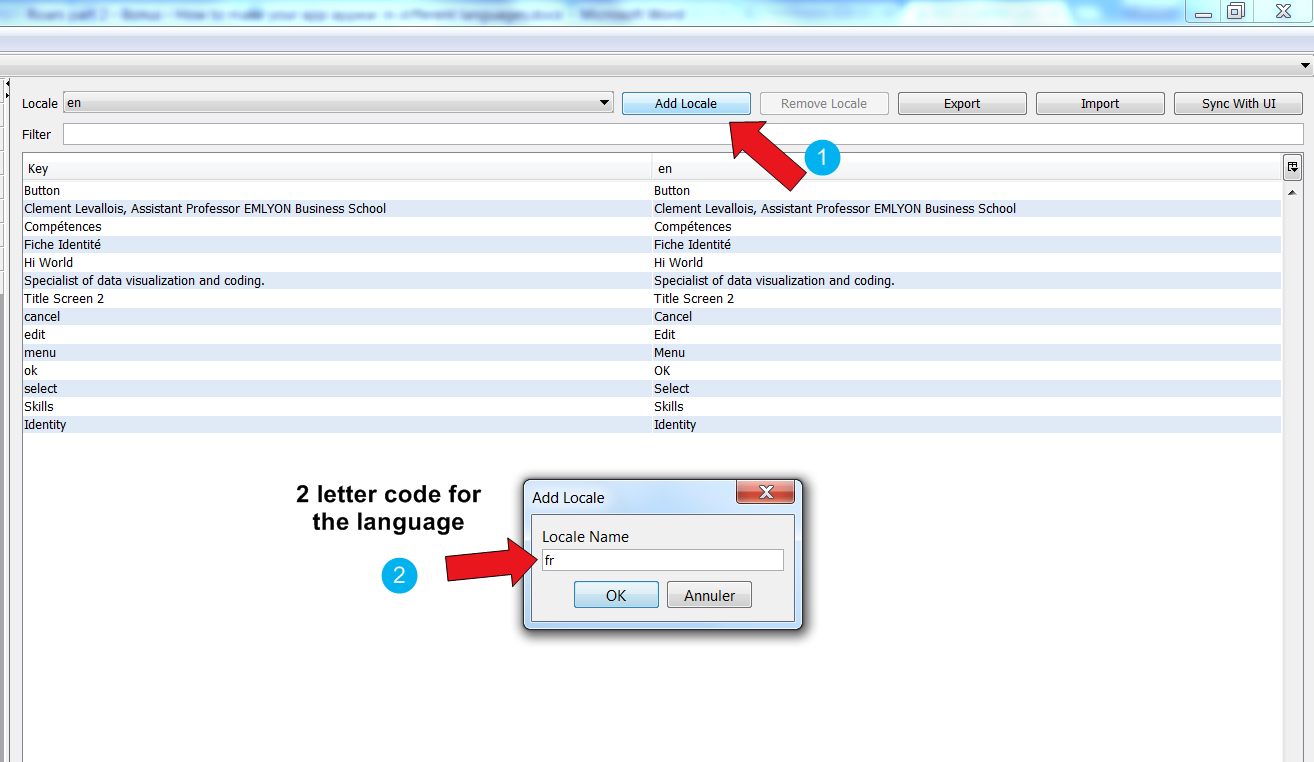
*How to make your app appear in different languages*

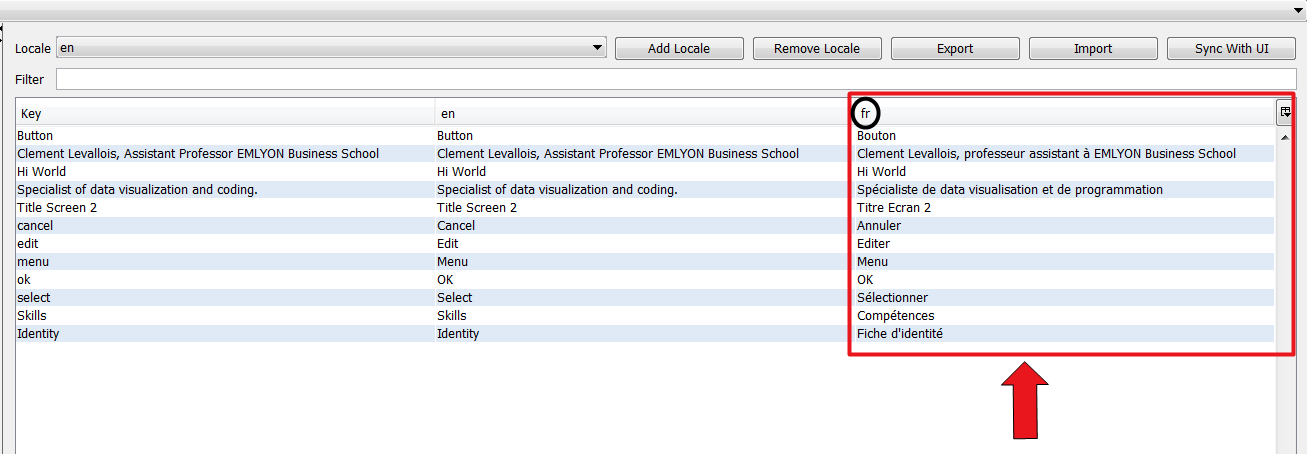


1. Now we have all the elements of text in our app appearing in a list. For each of element, we can create a text in a different language. To do that, just click on “Add Locale” and write the 2-letter symbol for the language (fr = French, de = German, …)

The list of all 2 letter symbols for all languages: <http://www.sitepoint.com/web-foundations/iso-2-letter-language-codes/>

**Here I show it in French**, “fr”:



1. A new column has been automatically created on the right (squared in red on the screenshot below). You can type the translations in French for all elements (or any language you have chosen if not French).

Level of difficulty:   
Estimated time: 15 mn

[**www.codapps.io**](http://www.codapps.io)by Clement Levallois **@codapps\_io** [**www.coursera.org/learn/codapps**](http://www.coursera.org/learn/codapps)

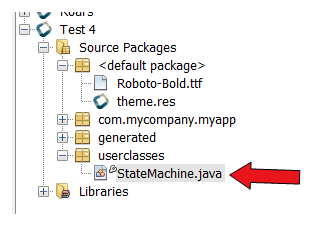


**Module 7 - BONUS**

*How to make your app appear in different languages*

1. That’s it, your app is ready to be displayed in English, or in French! But how can the app “know” when it should show English text, or French text?

First, go back to NetBeans, and double click on the file StateMachine.java to open it:



In this file, add these lines of code in the action “initVars”:

**This line of code should already be in your file. It means « execute everything inside the green curly brackets when the app starts »**

protected void initVars(Resources res) **{**

String languageOfThePhone;

**First, you create a box specialized in containing text.**

languageOfThePhone = Display.getInstance().getLocalizationManager().getLanguage();

**Then, we use a special action (just copy paste it…) which finds the language of the phone, and we put it in the box we have created.**

if (languageOfThePhone.equals("fr")) **{**

**If the lanuage of the phone, that we have put in the box, is « fr »…**

UIManager.getInstance().setBundle(res.getL10N("Translations for my app", "fr"));

**… then use this special action which makes your app use the texts from the column « fr » in « Translations for my app »**

**}**

else **{**

UIManager.getInstance().setBundle(res.getL10N("Translations for my app", "en"));

**… otherwise, it means the phone is not in French. Use the texts from the column « en » in « Translations for my app »**

**}**

**}**

Level of difficulty:   
Estimated time: 15 mn

1. Note: the lines of code above can simply copy pasted from here:

<https://github.com/seinecle/CodenameOneLocalization/blob/master/src/userclasses/StateMachine.java#L29>

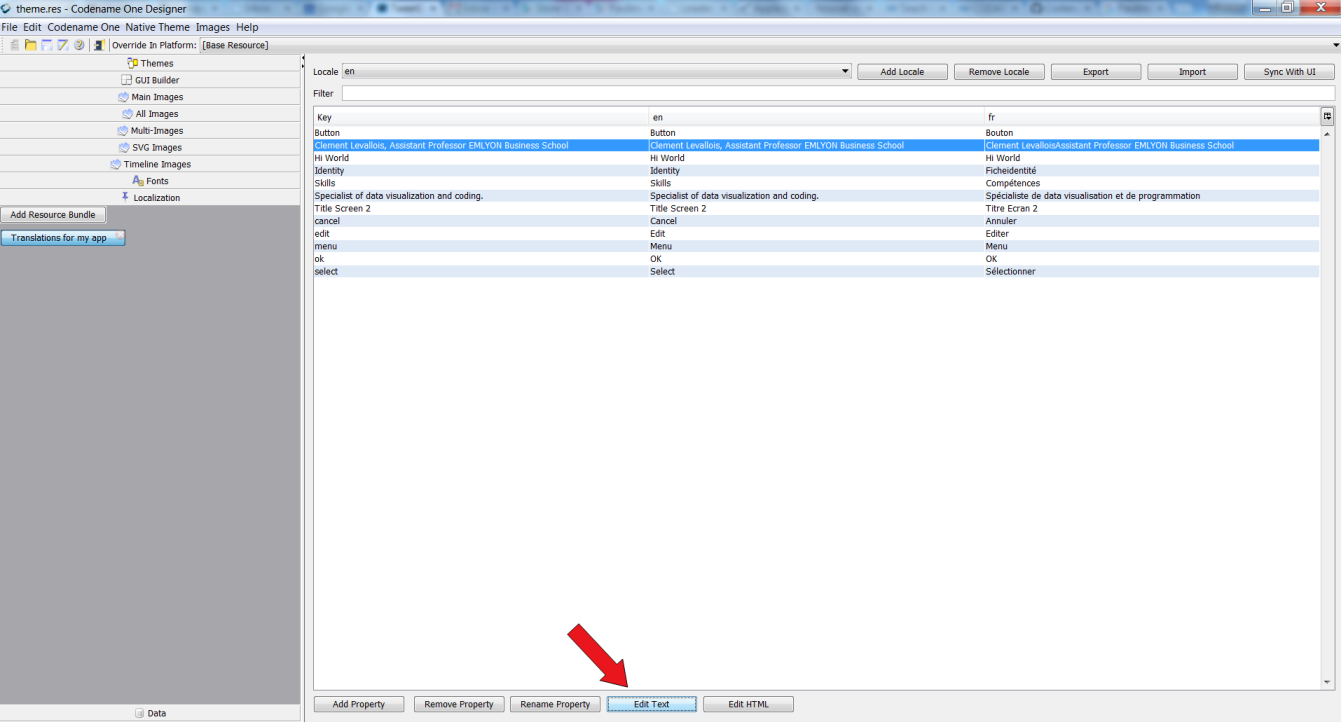
**Just make sure that you make no mistake on the line:**

UIManager.getInstance().setBundle(res.getL10N("Translations for my app", "fr"));

* I use “Translations for my app” because that’s how I named the Resource bundle in step 2 of this tutorial, but of course you should use any name you have chosen in step 2.

1. **The bonus of the bonus… how to have Labels with some text on several lines**

* Create a localization, as explained in steps 1 to 6
* Open the Localization menu, select the line where you want to have the text on several lines, and click on the “**Edit Text**” button:





[**www.codapps.io**](http://www.codapps.io)by Clement Levallois **@codapps\_io** [**www.coursera.org/learn/codapps**](http://www.coursera.org/learn/codapps)

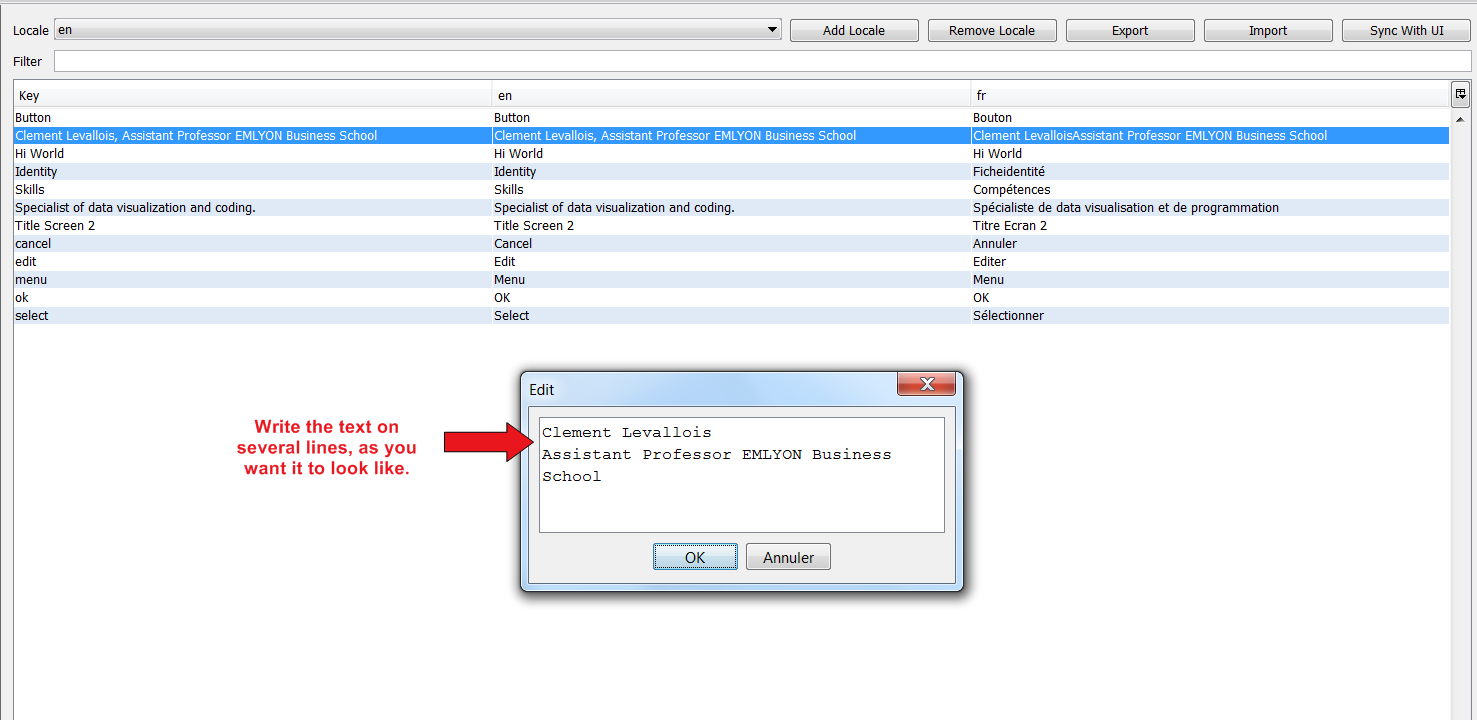


**Module 7 - BONUS**

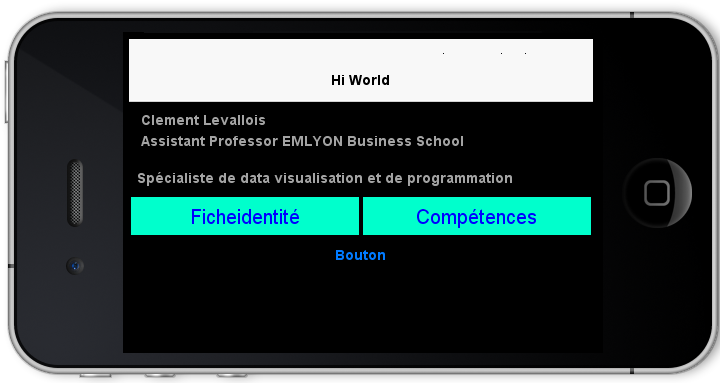
*How to make your app appear in different languages*

1. In the box that opens, you can write your text in several lines. To create a new line, just press the “Return” key on your keyboard (as usual).

Level of difficulty:   
Estimated time: 15 mn



1. Now you can have one Label with the text on 2 lines:





[**www.codapps.io**](http://www.codapps.io)by Clement Levallois **@codapps\_io** [**www.coursera.org/learn/codapps**](http://www.coursera.org/learn/codapps)



**Module 7 - BONUS**

*How to make your app appear in different languages*