

CENTRE FOR ENDANGERED LANGUAGES

SIKKIM UNIVERSITY

Meta Documentation

*How to compile an Android Dictionary application using
Dictionary App Builder*

Pabitra Chettri
Technical Assistant Cum Archivist

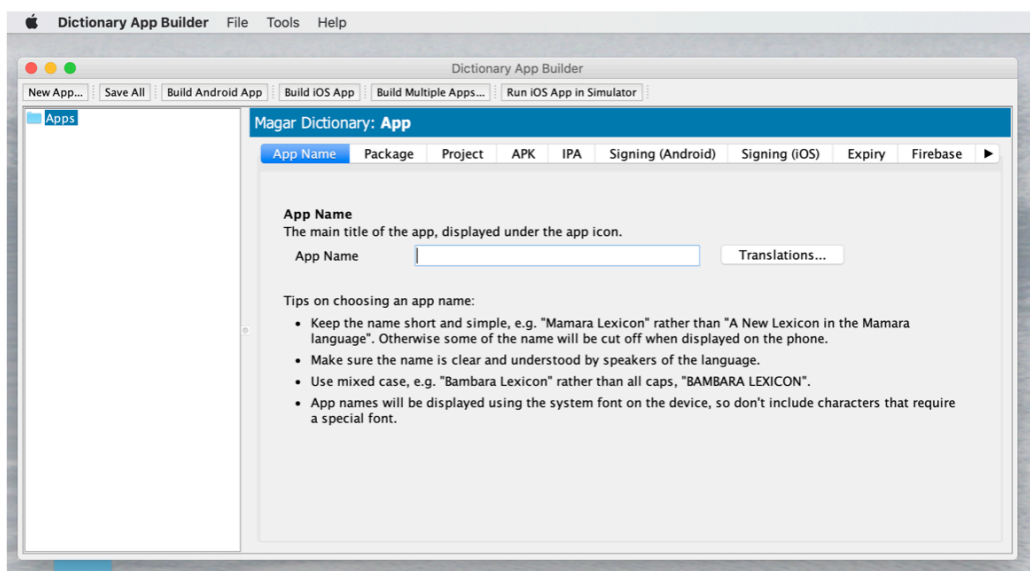


This work is licensed under a Creative Commons Attribution-NonCommercial-ShareAlike 4.0 International License. To view a copy of this license, visit <https://creativecommons.org/licenses/by-nc-sa/4.0/>.

How to build an Android Dictionary App(lication) using Dictionary App Builder

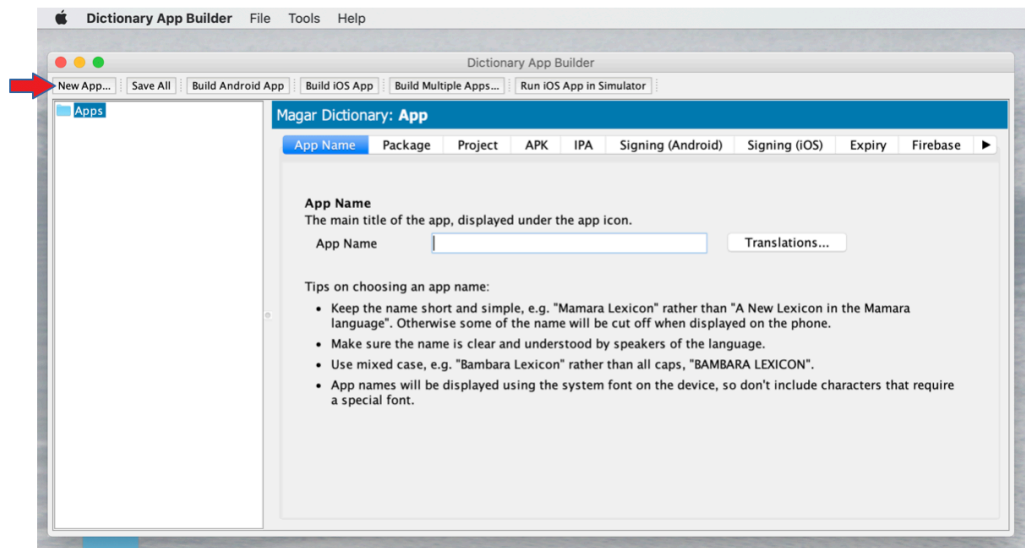
Dictionary App Builder is software that is used to build a customized dictionary app for the Android and iOS platforms which can be used on phones and tablets. The software can be downloaded from *software.sil.org*. Once the Dictionary App Builder is installed, certain series of steps has to be followed in order to build a dictionary for a language that one is working on.

The home page of Dictionary App Builder looks as shown in the figure below.



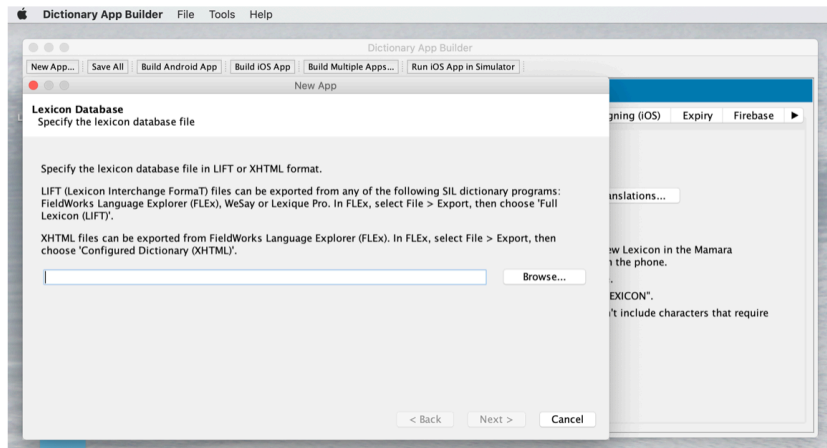
Steps for creating dictionary using Dictionary App Builder

Step 1: Click on New App menu on the toolbar shown on the home page of Dictionary App Builder.

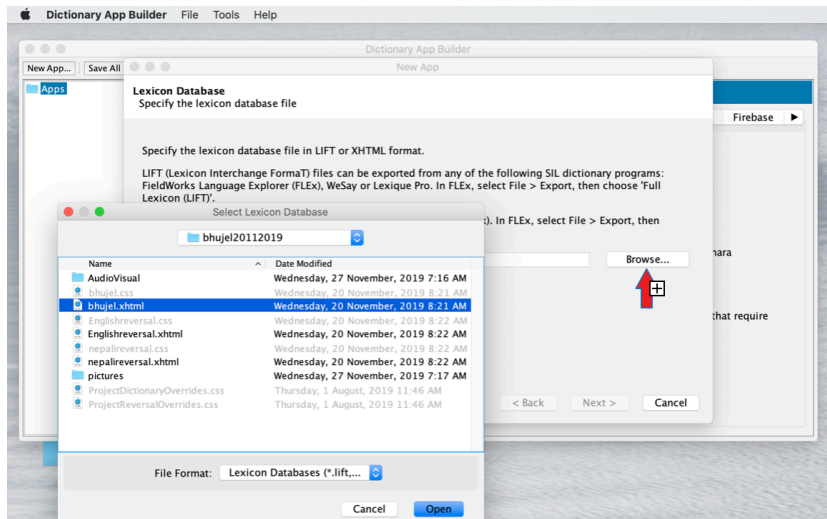


Meta Documentation

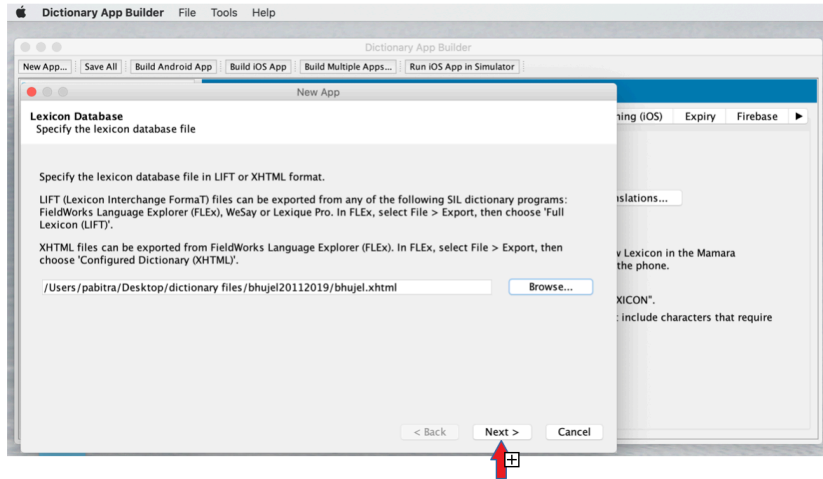
Step 2: New App dialogue box opens up where the lexicon database file has to be specified.



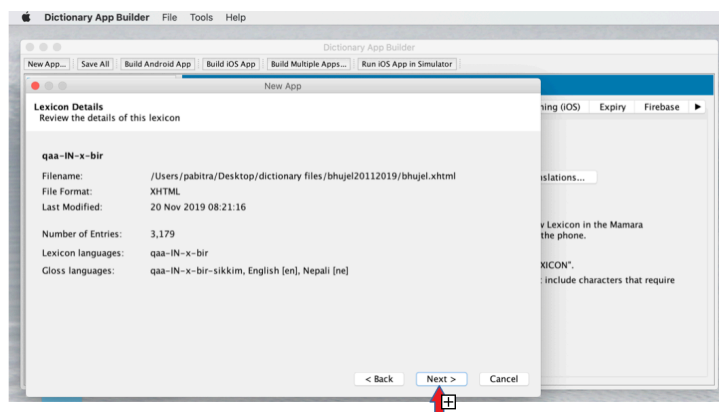
Step 3: Browse and select the LIFT or XHTML file that has been exported from FLEX, WeSay or Lexi-Pro.



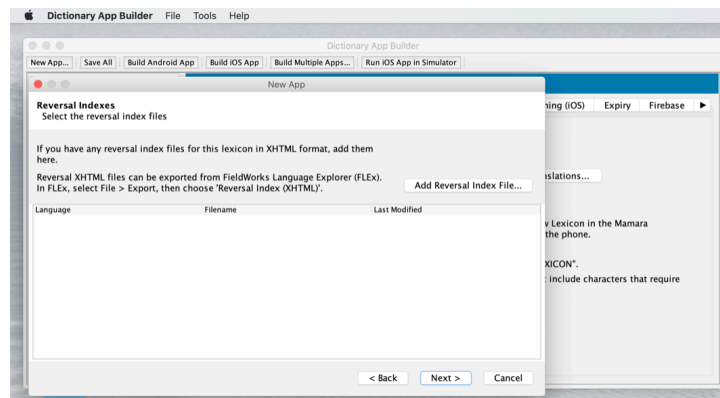
Step 4: After selecting the required file click on the Next button.



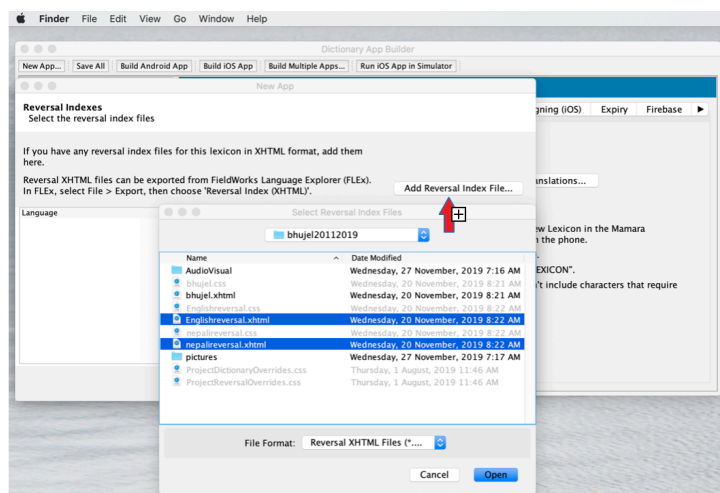
Step 5: After clicking the Next button the Lexicon Details window will be displayed. This dialogue box specifies the Filename, File Format, the Number of Entries in the lexicon, the Lexicon language, Gloss language. Review the details of the file and if it matches the source file you can move ahead and click on the Next button otherwise select the correct exported file.



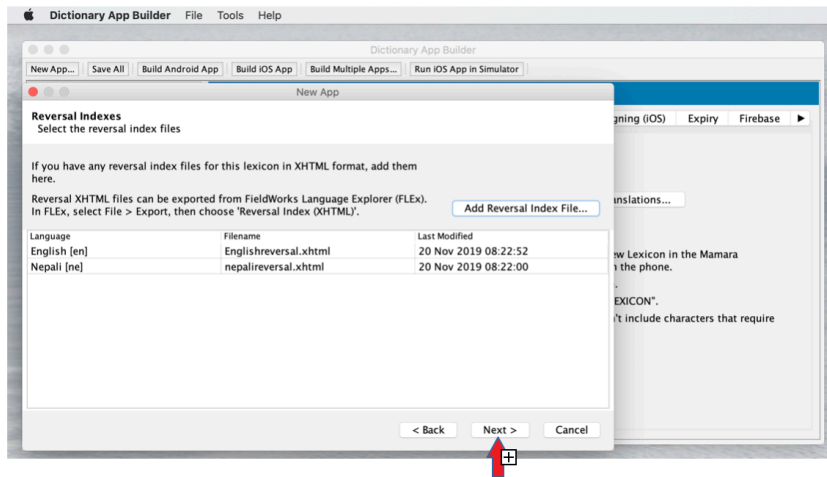
Step 6: If you are using an XHTML file to build the app one needs to add the Reversal Index file separately. The Reversal Index has to be exported separately from FLEx. The New App dialog box allows you to add the Reversal Index file. In the New App dialog box add the Reversal Index File that has been exported from the FLEx.



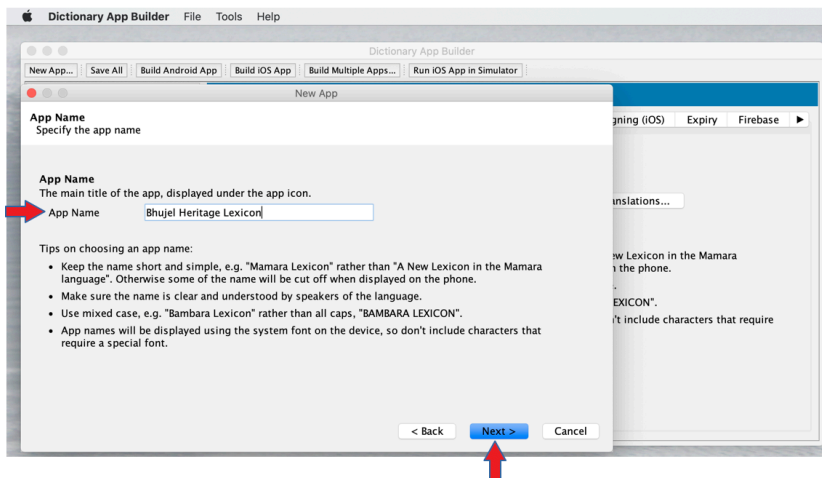
Step 7: Click on Add Reversal Index File button on the New App dialog box. Choose the reversal file / files exported from the FLEx that are to be added in Dictionary App Builder.



Step 8: Once the Reversal Index File is added the file/files should be visible on the New App dialog box as shown below. Now you can proceed and click on Next button.



Step 9: On the New App dialog box provide the App Name on the slot as per the instruction in the dialog box (Tips for naming the App is also provided). Click on the Next button once it is done.



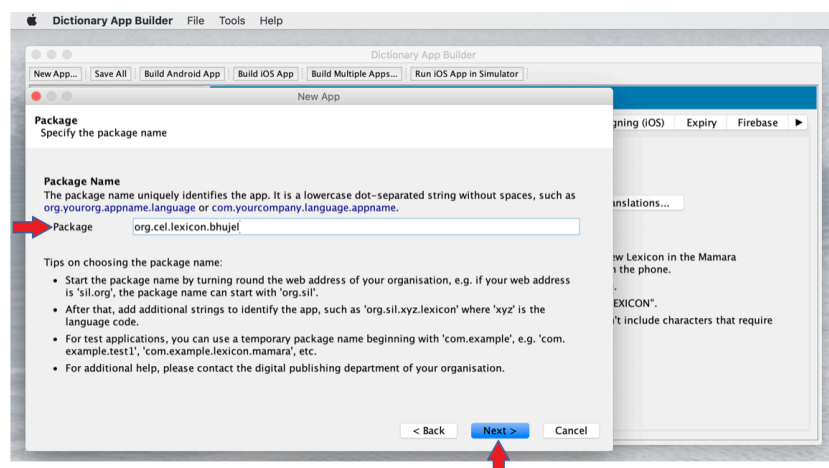
Step 10: After the App name, specify the Package name. The package name is used to uniquely identify the app. The Package name should be *lower case dot separated string without space*. It can be as shown in the example below.

org.yourorg.appname.language OR *com.yourcompany.language.appname*

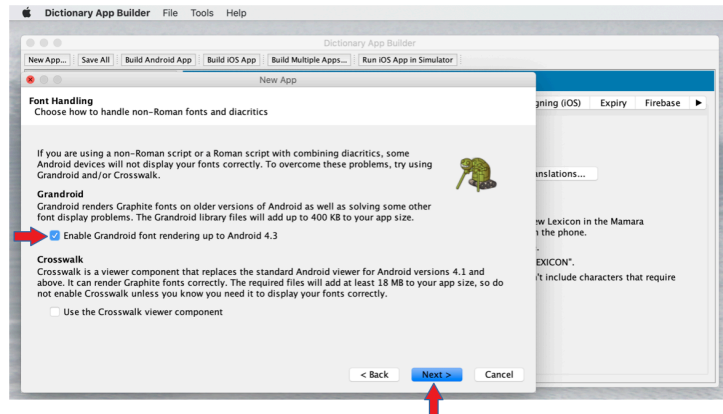
org.cel.lexicon.sherpa OR *com.cel.sherpa.lexicon*

org.cel.dictionary.magar OR *com.cel.magar.dictionary*

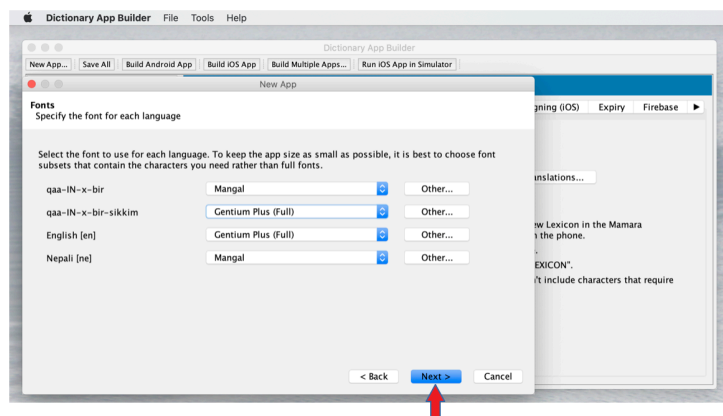
Once the Package name is assigned click on the Next button.



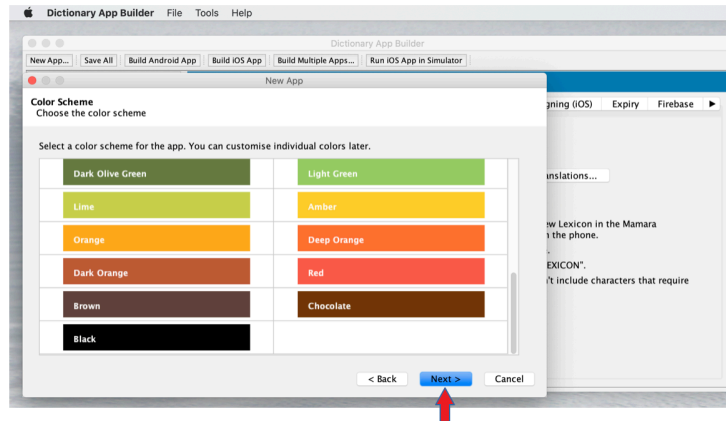
Step 11: The next step **Font Handling** allows one to handle non-Roman fonts and diacritics. To do this enable **Grandroid** and **Crosswalk** option. Check on the required features and then click on the **Next** button.



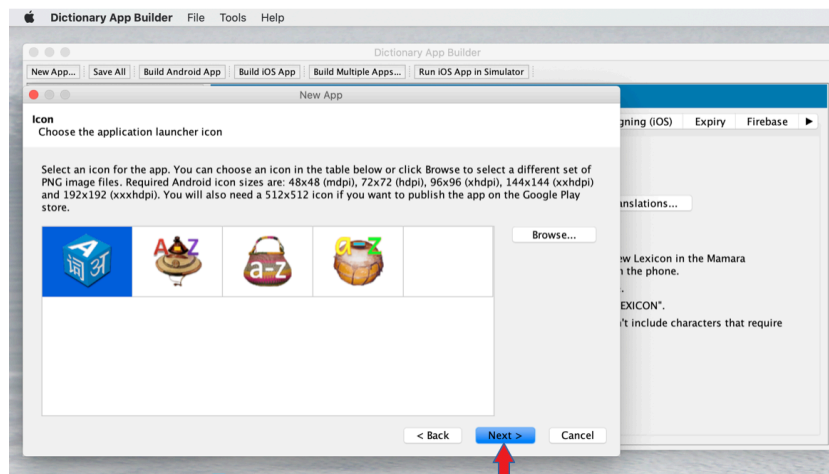
Step 12: The next dialog box is to specify the **Fonts** used for the language(s) both the vernacular and analysis languages that were used while making entries in **FLEX**. One must choose compatible fonts from the list. For those fonts that are not visible on the list one can click on the other button to specify the required font file. Once the appropriate fonts are being selected click on the **Next** button.



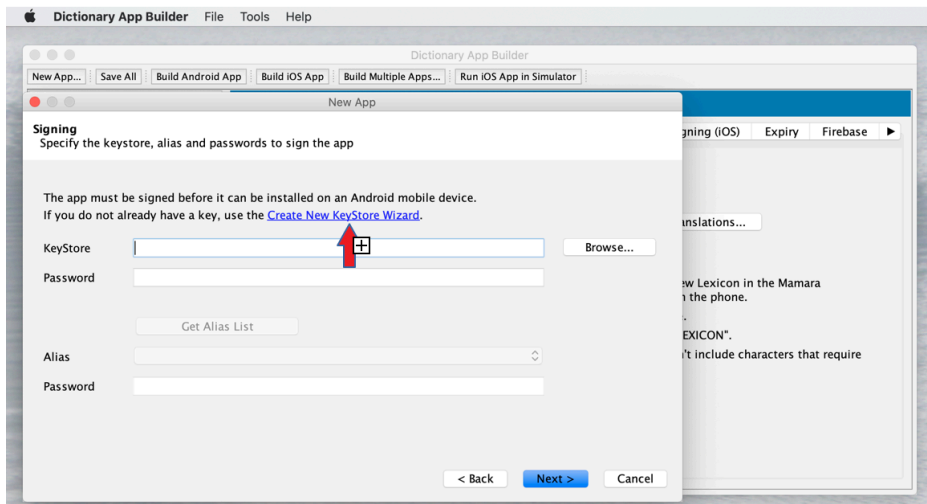
Step 13: The next step choose the Color Scheme for the app and click on the Next button.



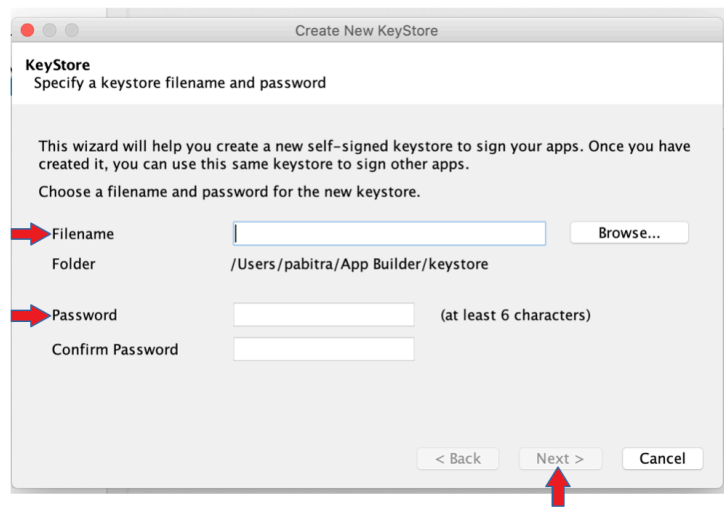
Step 14: Select the Icon for the App. One can choose from the available sets of icons or can use custom icons. The custom icons can be selected from the Browse option. However, the custom icons must be of the appropriate size specified in the window. Once the desired icon is selected, click on the Next button.



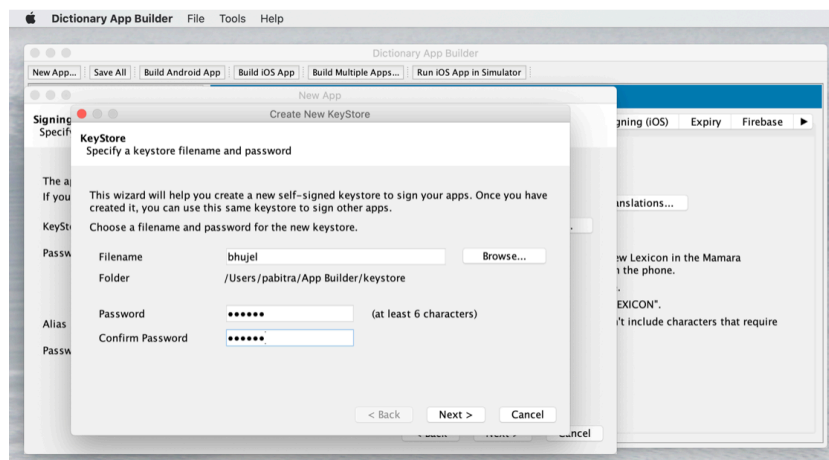
Step 15: The next step is to specify the Keystore and Passwords to encrypt the app. If one has already created a key file then simply browse and select it. If you are creating a new keystore then click on Create New Keystore Wizard. It is a onetime process which means once you have created a keystore and password for a language then it can be used any number of times whenever you modify and improve your App. One does not need to create the key again and again. However ensure to remember the password.



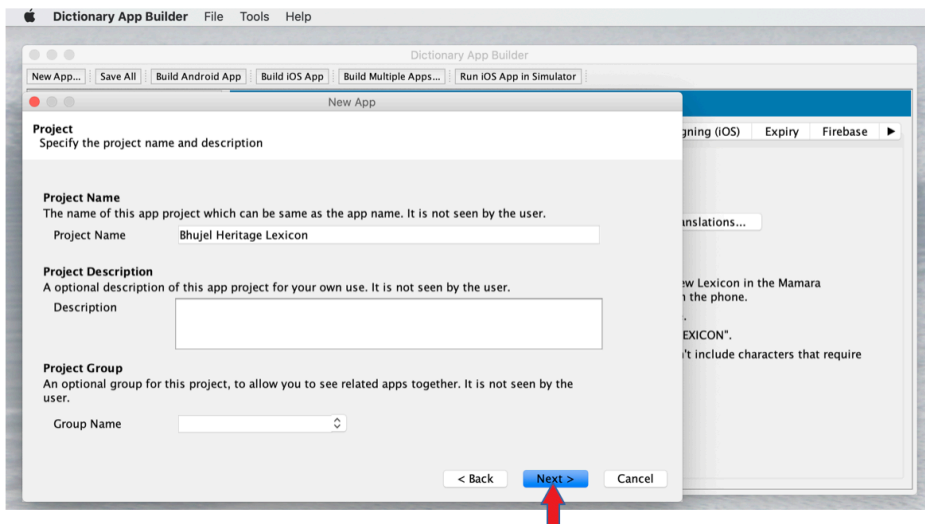
Step 16: After clicking on Create New Keystore Wizard, a Create New Keystore dialog box will appear. In this fill the keystore filename and password (at least 6 characters) for the key. Once done click on the Next button.



Step 17: Now you can find the recently created key selected in the Keystore window. Click on the Next button



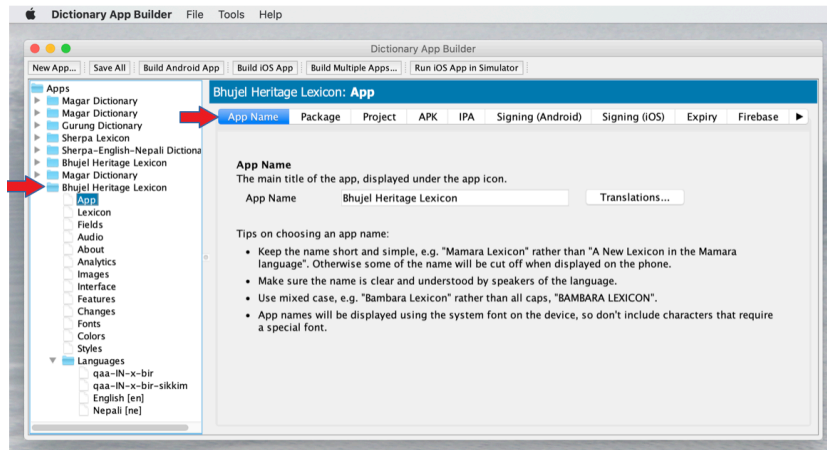
Step 18: In this step specify the Project name and a Project Description. These fields are not visible to the user when the app is installed. Here the description of the project is optional. You can give the details like who created the app, date on which app was created and etc. Click on the Next button after filling up the details.



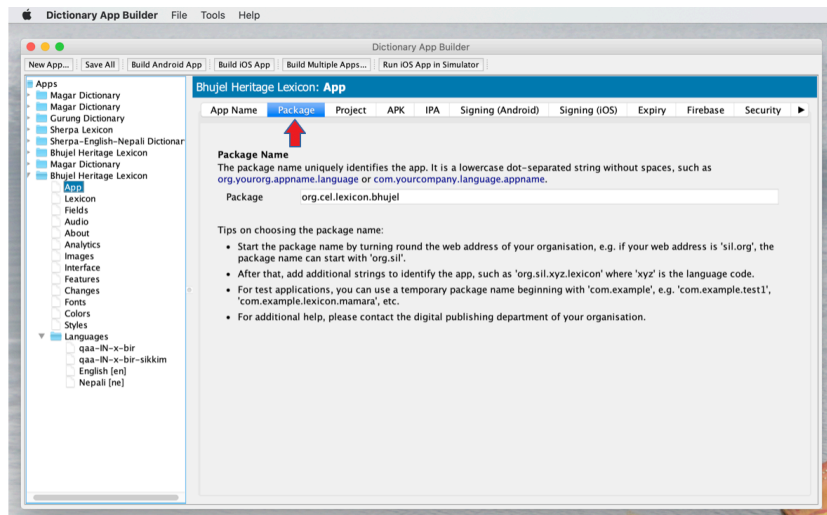
Step 19: Once all of the above mentioned process is successfully completed a new app project will be created in the Dictionary App Builder. The window below shows the various details. By default the window shows the App Name that you have given in step 5. All the previously filled in details can be accessed through various tabs. These tabs are App Name, Package,

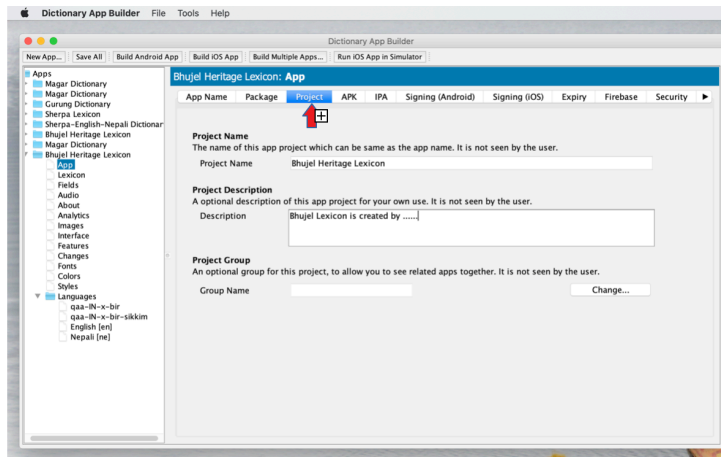
Meta Documentation

Project, APK, IPA, Signing (Android), Signing (IOS), Expiry, Firebase, Security and Permission.

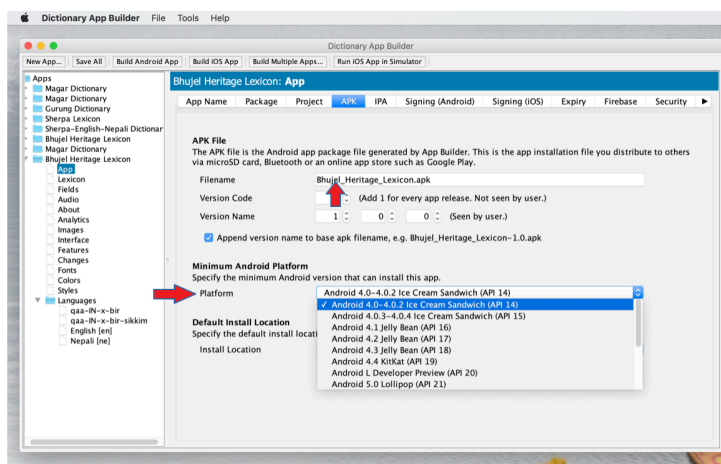


Step 20: Use the tabs Package and Project to check the information specified and update when necessary.

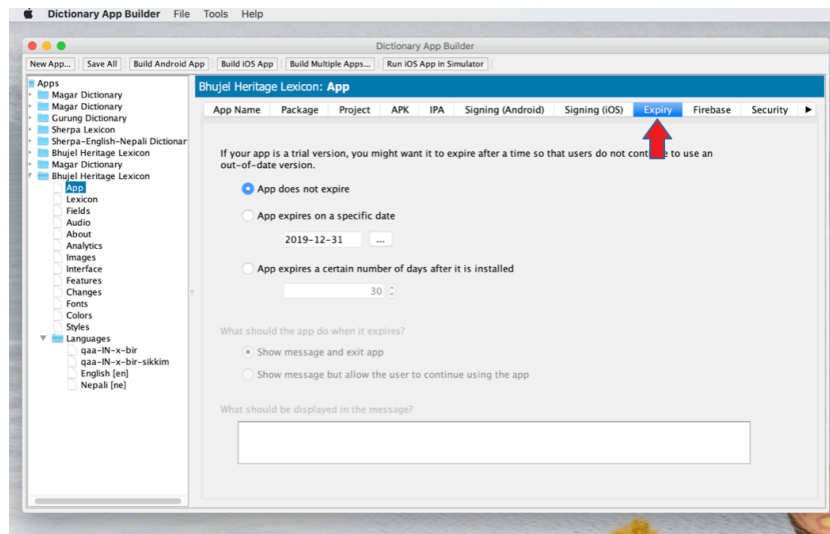




Step 21: An APK file is an Android app package file generated by the Dictionary App Builder. This file is used for app installation. It can be distributed to other users via Bluetooth, SD card or shared by other means. The APK tab shows the automatically generated Filename. One can update the filename when it is necessary. One must select the minimum Android version to which the app can be installed. The minimum Android Platform that is available is Android 2.3-3.3.2 Gingerbread (API 9).

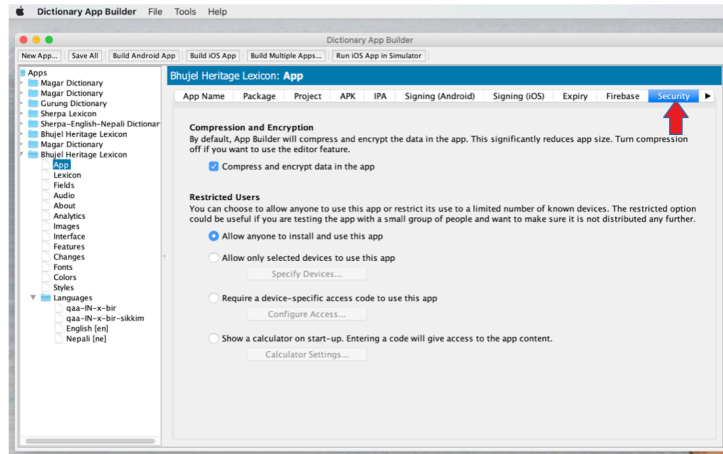


Step 22: The expiry of the App being developed can be decided by the creator. The Expiry tab is used for this. Three options are available for it. Choose the appropriate option and enable the checkbox to make it active.

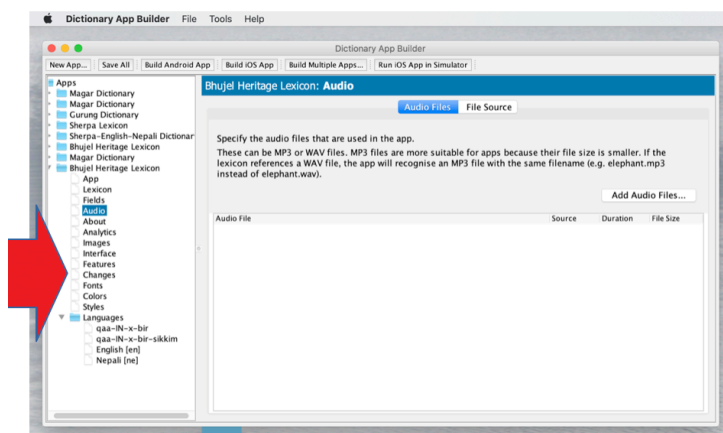


Step 23: Next to Expiry is the Security tab. When clicked on it the tab shows two options Compression and Encryption and Restricted Users. The Compression and Encryption of the data is by default enabled. One can choose to disable the option if required. The users of the app can also be restricted. One can allow anyone to install and use the app or can allow only the selected devices to use the app for which you need to specify the devices by entering the device name followed by the serial number available for the particular devices that can be retrieve from the about phone or about tablet page available in the settings of various mobiles and tablets. The restriction can also be made using device specific access code

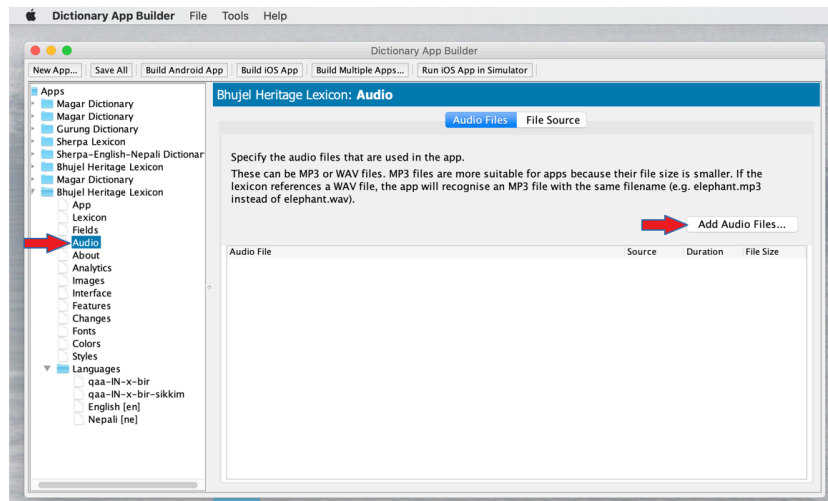
to use the app.



Step 24: On the left pane of the Dictionary App Builder one can navigate to other options in the project. You may have noticed that in this panel the project is listed under the main folder named Apps. Each project contains several options they are: App, Lexicon, Fields, Audio, About , Analytics, Images, Interface, Features, Changes, Fonts, Colors and Styles. Along with these a sub folder called Languages which list the languages used in the App. All these features allows us to customise the App.

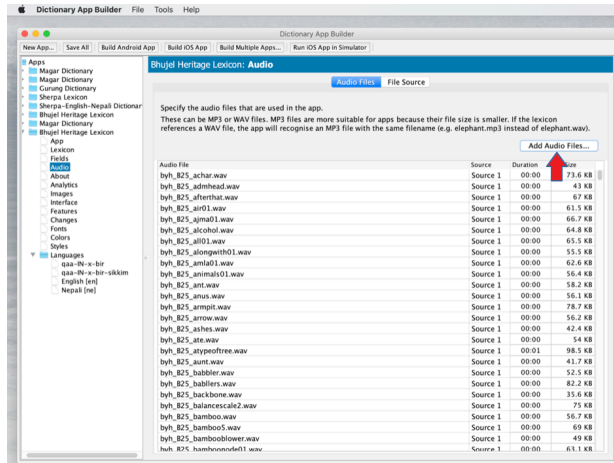


Step 25: The dictionary being built will contains thousands of words and for each word if audio can also be inserted then the dictionary becomes more attractive and it becomes easier for users to learn the pronunciations as well and becomes more interactive. For this click on the Audio option displayed on left pane of the window. Once you click on the Audio option the page as shown below will appear. In this page you can add audio by using the Add Audio Files button.

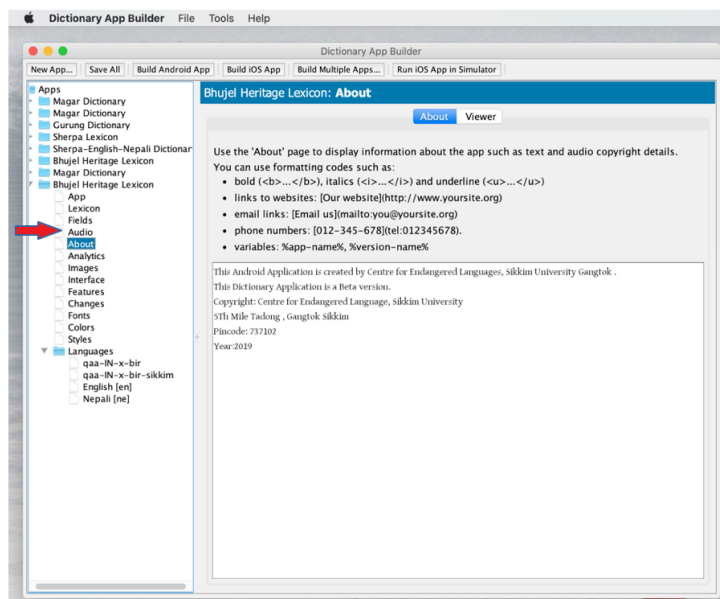


Step 26: After clicking on the Add Audio Files button, browse and select all the audio files that has been saved in the same folder where the LIFT or XHTML file was exported. Once all the audio files are chosen the window will be in the form as shown below. By default all the audios added to the

App will be packed inside the Apk file.

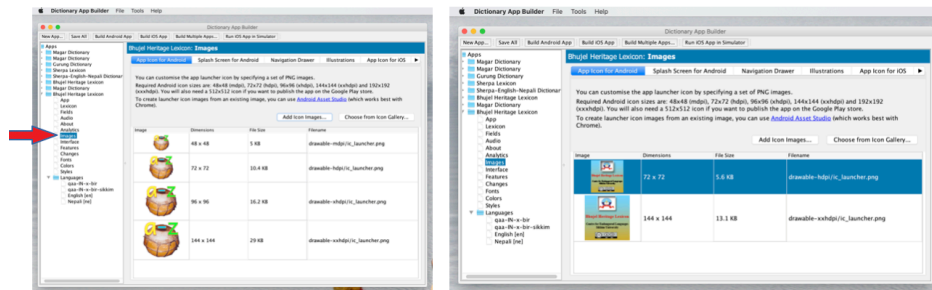


Step 27: The details about the application that is being developed can be given to the users to know more about the App. To do this click on the About option on the left pane of the window. In the about page you can give detailed information about the App.

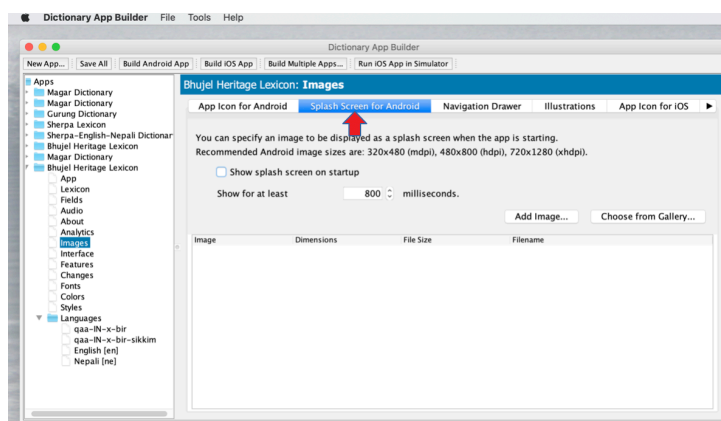


Meta Documentation

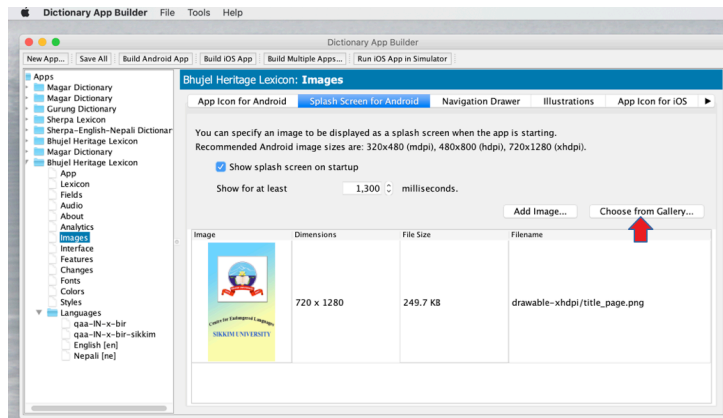
Step 28: The icon that has been selected for the app (step 14) can be customised according to the need. For this, click on Images option appearing on the left pane of the window. From the images page select the size that you think is suitable for the app. One can either choose from the default icon image that is available or can choose the icons that one has created of their own.



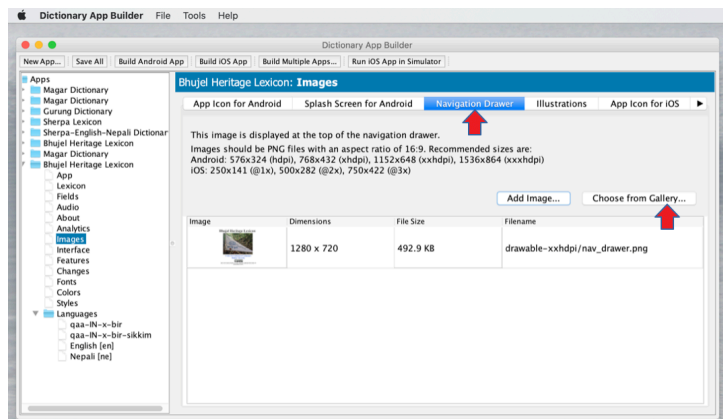
Step 29: One can add an image to the Splash Screen of the App that will be displayed whenever the App is starting and can also specify for how long to display the image on the screen. For this click on the Splash Screen for Android tab.



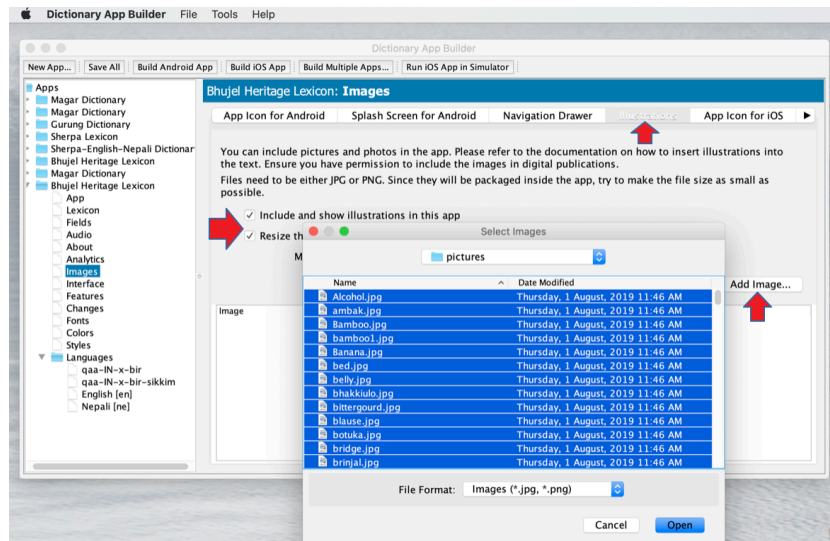
Step 30: After clicking on the Splash Screen for Android tab, browse the image for splash screen from the location where the image is stored by clicking on Choose from Gallery option and check on the box stating “Show splash screen on startup” and then set the timings as well.



Step 31: For adding image to the Navigation Drawer click on the Navigation Drawer tab and browse the image from the location where it is stored in the same way as done with the splash screen.

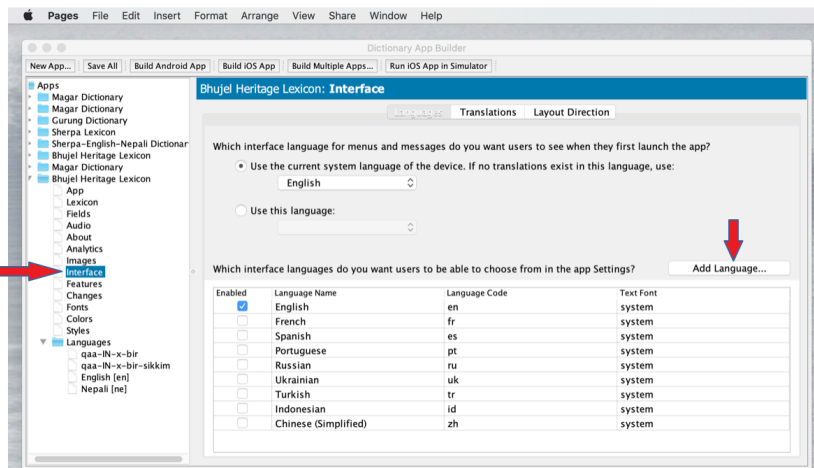


Step 32: The Dictionary will be more interactive if it contains images for the entries. So for this purpose one can use the Illustrations tab. The Illustration tab allows you to include illustration and images to the App and control their size. Check Include and show Illustration in this app to include the images. Then click on Add Image option this will let you browse the location where the images for each entries were saved. The image folder will also be exported from the FLEx automatically whenever the lexicon database file is exported. Once the images are added check on the resize options and set the desired width for the images.

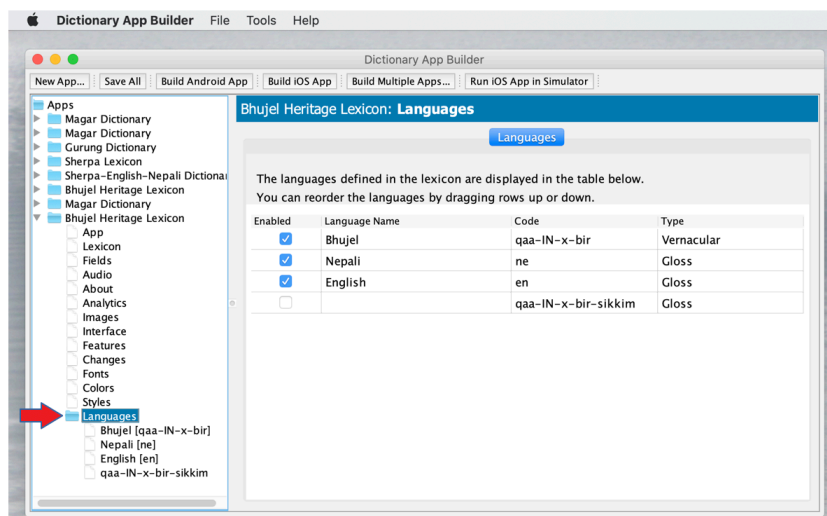


Step 33: The language for menus and messages can also be customised. Click on the Interface option available on the left pane of the window to do this. The Interface page allows you to choose the language that you want the user to see for various menus and messages. One can choose from the

available list of languages or can add new language by clicking on the Add Language button.

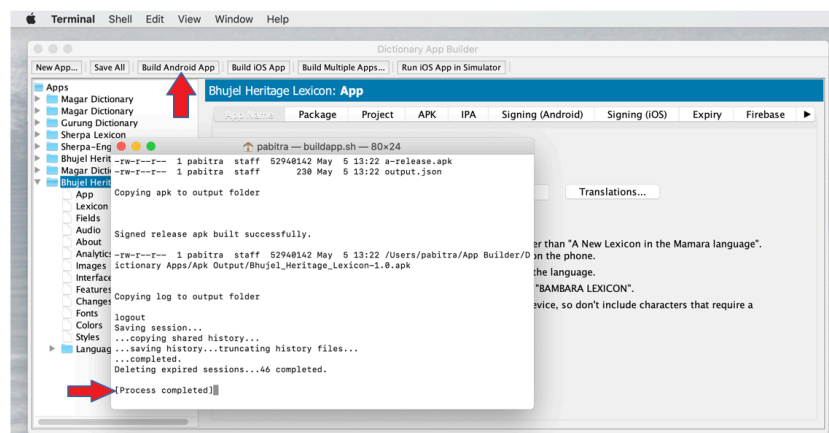


Step 34: The Languages option on the left pane of the window allows you to control the order of the dictionary. The languages defined in the lexicon will be displayed in the window. It can be reordered just by dragging the rows up or down and can even be unchecked if not required.

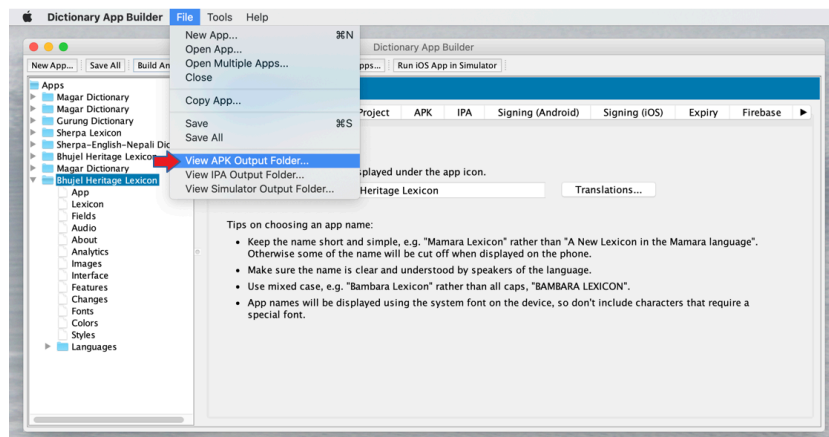


Meta Documentation

Step 35: These are the various steps that are needed to be followed for building the apk file. When all the features are selected and all the mandatory details are entered the final step is to build an apk file. For this click on **Build Android App**. Once you click on it build program runs up and after it is completed it will display a message “Process completed”.



Now on the main window of Dictionary App Builder, you can go to: **File > View APK Output Folder** to get the apk installation file for the dictionary.



The figure below shows the folder which contain the apk installation files that can now be installed in your android phones and tablets.

