# Key

## ✔ Done

## ➀ Planned for first version

## ➁ Planned for second release

# Installer

## ✔Install sample data so evaluators can try it out fast.

## ➁ Update old version without requiring uninstalling first.

## ➁ Allow user to choose an installation location other than “Program Files”.

# WeSay Setup Application

## ✔Create a new project

## ✔Launch a project in WeSay

## Task Setup

### ✔Control which tasks to enable for the user at this time

### ✔Provide several ready-to-use tasks: “Add Meanings”, “Add Examples”, “Collect Word List”, “Collect Words By Semantic Domains”.

### ➀ Control which fields are shown for each task. Until then they can be tweaked by hand in the tasks.xml file.

## Field Setup

### ✔Control which writing systems are shown for each field, and in what order.

### ➀ Allow user to create custom fields in the Setup App. Until then create by editing the tasks.xml file directly.

## Writing Systems

### ✔Specify a font for the writing system

### ✔Mark if the writing system uses right-to-left text.

### ✔Enter the name of a keyboard to use when in fields of this writing system. The keyboard can be either a system one or [Keyman](http://www.tavultesoft.com/keyman/).

### ➁ Support alternative keyboard on Linux, such as [kmfl](http://kmfl.sourceforge.net/).

### ✔Support complex non-roman fonts on Windows

### ➁ Support complex non-roman fonts on Linux

### ✔Allow multiple writing systems per field.

### ✔Specify which writing systems appear in which tasks.

### ✔Changing the writing system id automatically updates the lexicon xml.

### ➀ Specify an on-screen abbreviation for the writing system, separately from the writing system id.

### Specify sort order other than standard Unicode order.

### ➀ Specify a language which Windows knows about in order to sort like that language.

### ➁ Use a library like [ICU](http://www-306.ibm.com/software/globalization/icu/index.jsp) to expand the number of build-in sorting systems.

### ➁ Provide a way to define custom sorting systems.

### ➁ Store the information for a writing system in its own xml file, conforming to [LDML](http://www.openi18n.org/specs/ldml/1.0/ldml-spec.htm) with some SIL enhancements: default font, keyboards. Allow reuse of this data across applications by sharing these files.

## Option Lists

### ✔Allow custom fields to be limited to a list of options.

### ➀ Edit option lists in separate program (e.g. parts of speech). Currently xml-only.

# WeSay

## Home Tab

### ✔Tell user how many word they have so far

### Show histogram indicating progress over time

### ✔List all the tasks the advisor has enabled using the Setup application.

### ✔At the top, show and highlight the task the user was in previously, to remind him where he was.

### ➀Next to each task show how many words/senses still need that task. (Currently, this number is sometimes inaccurate).

### ✔Right-clicking enables quickly opening the project in the Setup application.

## Lexical Model (what is supported)

### ✔Supported *objects* are Word, Sense, and Example

### Support LIFT’s *relation* object

### Support LIFT’s *phonetic* object.

### Support LIFT’s *variant* object.

### Support LIFT’s *reversal* object.

### ✔Words or phrases, in one or more writing systems.

### ✔Words can have one or more Senses

### ✔Senses can have one or more examples

### ✔Examples have a single translation (though it can be in multiple writing systems).

## Missing Fields Tasks

### ✔Show list of words that are missing the field

### ✔Move words to the “completed” list when the field is filled in

### ➀ Show other fields for context, but read-only

## Dictionary Task

### ✔Add New Word

### ✔Delete This Word

### ✔Show list of all words down the side.

### ✔Change the writing system of the list via a drop-down above it, so that you can view, for example, all the English glosses, or all the words in a particular writing system.

### ✔Type-ahead search box.

### ✔Search box uses approximate matching to find word even without knowing the exact spelling.

### ✔Show dictionary output preview of this word, at the top.

### ✔Underline the part of the dictionary output corresponding to the field which has the cursor.

### ✔Always provide a blank line to add more information, such as another meaning (sense), or another example sentence. *(This is under review… does it make the program seem like it is never satisfied?)*

### ✔Delete a sense or example by simply erasing its contents.

### ✔Provide a star button for indicating something special about a field, such as the need to review it later.

### ➀ Support editing of collection traits (e.g. Semantic Domains).

## Collect Word List

### ✔Collect words in the target language by displaying one word at a time in a foreign language, and asking the user to translate it into one or more words in his language.

### ✔Words are entered into the dictionary, with the foreign word listed as a gloss.

### ✔If the word is already in the dictionary, a new sense is created, with the foreign word as its gloss.

### ➀ Addition of Words animated to reinforce what’s going on.

## Semantic Domain Task

### ✔ Addition of Words animated to reinforce what’s going on.

## Actions

### ➀ Provide a way to add plugins which do some action.

### ➁ Email advisor.

### ➁ Submit to Subversion repository.

### ➀ Print draft dictionary.

### ➀ Print progress report.

### ➁ Open dictionary in Lexique Pro.

# Notes

## Since WeSay seeks to shield the user from the file system, there is no “Open…” command. WeSay can be started with a particular project in these ways:

### By double clicking on the LIFT file

### Via Project:”Open this project in Wesay” in the WeSay Setup program

### Via a shortcut or command-line parameter which gives the path to the LIFT file

### By running WeSay, in which case it opens the last project it was run with.

## The setup program places new project in the “My Documents” folder, but you can move them anywhere.

# Keyboard Shortcuts

## Standard Windows ones

### ✔ Tab: move to next field

### ✔Shift-Tab: move to previous field

### ✔Ctrl+PageDown Move to Next Tab Page

### ✔Ctrl+PageUp Move to Previous Tab Page

### ✔Alt+Down Open up a combo-box (e.g. Part of Speech)

## Dictionary Task

### ✔Alt+N: New Word

### ✔Alt+D: Delete Word

### ➀ Ctrl+F: Move to find fox

## Collect By Semantic Domain

### ✔Enter: Add word to list of words with the displayed semantic domain

### ✔Page Down: Move to the next question or, if out of questions, to the next topic.

## Collect By Semantic Domain

### ✔Enter: Add word to list of words which are equivalent to the displayed foreign word.

### ✔Page Down: Move to the next word to translate.

## Missing field tasks (e.g. Add Meanings)

### ✔Page Down: Move to the next word which is missing this field