

# Use Excel to Build CommCare Forms



## Warning!

There are many XLSForm [advanced formatting options](#) that will not work with CommCare. Please review the Known Limitations section below carefully before building forms using XLSForms!

Have a very long survey that you'd rather put into an Excel file before uploading in the [form designer](#)? You can use [XLSForm](#), a free, online form-building tools that allow you to design your form in an Excel file, and then copy the created XML code into the CommCareHQ form designer. You can work on the Excel file offline and then upload it later.

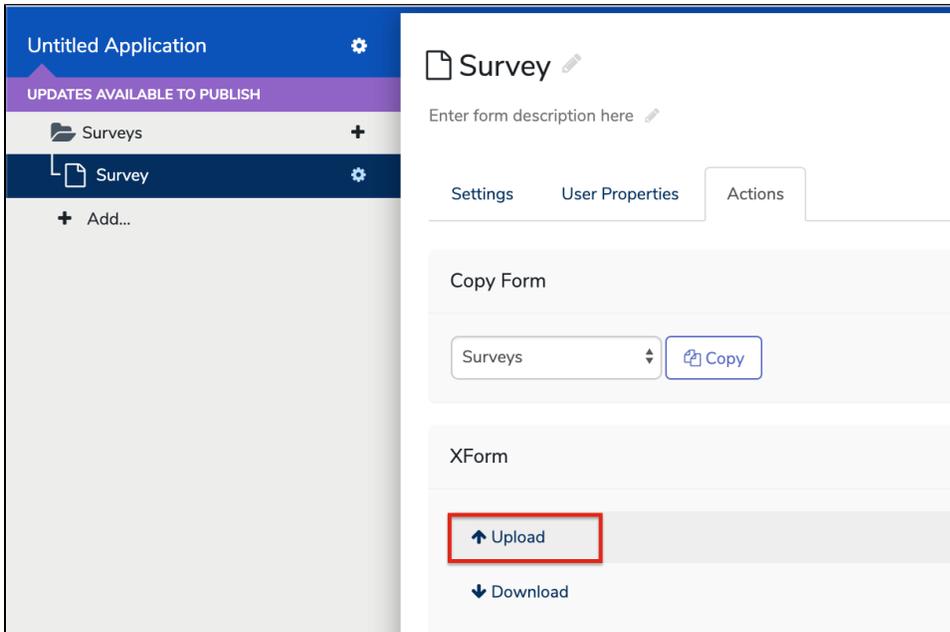
You can see detailed instructions on how to set up the Excel file [here](#).

## Overview

Build your form in Excel (following the instructions on <http://opendatakit.org/use/xlsform/>) and then upload it to XLSForm (<http://opendatakit.org/xiframe/>). You can then download the XForm and upload it into our form builder on HQ.

As you build your form you can test it online using the validation tool: <https://opendatakit.org/use/validate/>

1. To load into CommCare, download the xform, a file that will end in .xml
2. Once in the app builder on CommCareHQ, create a new blank form. You can then upload the xform directly on the form settings page by clicking on the gear icon for the respective form, then selecting the "Actions" tab and "Upload".



## Notes while building your form

- We strongly recommend that you test this with a short form before building a huge form so that you are comfortable with the process.
- If you are familiar with XML, you may notice some differences in syntax between the XML generated and CommCare's language – Don't worry! Noticeable differences are validation writing and writing for display conditions (what XLSForm calls "relevancy"). As all of these tools are based on the same standard, CommCare HQ will automatically fix the minor differences from FormHub syntax to meet our syntax requirements.

Example: In CommCare's Form Builder, a display condition might look like: `/data/pizza_fan = 'no'` while in Formhub it would look like: `$(pizza_fan) = 'no'` - these will be updated when the xml file is generated

- To avoid issues with itext and how CommCare deals with languages make sure to specify the language in the header. For example, if your app is only in English then make sure to put "label:en" instead of just "label" - if you do CommCare will add two languages (default and English).
- To keep using "data" as the ID of each form (the CommCare standard), please be sure to name your excel file "data". CommCare may not recognize the "settings" tab of the Excel file described XLSForm.
- Labels (called "notes" in the other tools) are often treated as text questions when the form is uploaded to CommCare. You can change the question type from Label to Text in the CommCare form builder. This is a known issue and will eventually be fixed. However, if you use the question type "acknowledge" then CommCare will interpret that as a label with the checkbox confirmation turned on.

- You can test your forms online using <https://opendatakit.org/use/validate/>

## Using non-Latin languages and alphabets in the XLSform-XML process

If you choose to use XLSform's ability to handle multiple languages, by reading the XLSforms documentation and adding additional columns to contain the additional language content, please note the following:

- The XLSform documentation tells you to create additional columns and give those new columns headers like **label::English**. However, if your purpose is to eventually paste that XML into CommCare, then you need to be careful about what the title of the language you use in those column headers. For commcare use, you should not use the full language name like "English," but instead must use the "short language codes," such as **en** and **tha**, that are shown in the little "Languages" pull-down menu in CommCare's left nav bar. Therefore your additional column headers will look like **label::en** or **required\_message::tha**.
- If your additional language content uses alphabets other than the latin character set, you will have to be careful about the way that you handle the process of copying your finally converted XML file into CommCare. Normally, many users would simply double-click the XML file that they had downloaded from the converter, select-all and copy, and then paste that XML into the CommCare forms editor. However, on Windows, it is likely that the default application that loads when you double-click the XML document will not display the non-latin character set correctly, and will therefore paste as gibberish into your CommCare form. If this happens, then, when you download the XML document from the converter, open it in a text-editing application that is more able to handle non-latin alphabets (one example is the Notepad++ app for Windows.) This app will correctly handle the non-latin characters in the XML, and then you can successfully cut-and-paste from that app into the CommCare form editor.

## Known Limitations

There are some things that don't work well when uploading a Formhub form to CommCare; these are active efforts to make these more compatible

- You may get an error message when uploading your form that CommCare xmlNs will be added. You can ignore this message. If, once inside the Form Builder, you see a persistent error message about the Form Name make any minor change in the form, save, and refresh the page.
- When first opening your form to edit you may see a message that says "id is not defined." Just click "ok" and then make any minor change in the form, save, and refresh the page.
- If you have multiple languages in your Excel form but not in your CommCare app, any additional languages that are not already in your app will be deleted. Before uploading anything go to the application -> languages and make sure all languages are added. On the languages page you will see that each language has a 2 or 3 letter code (English = en; Hindi = hin). In your Excel file you must use those codes in the headers. So for English you would use "label::en" as the header for the label column on both tabs of the Excel file. These are case sensitive. Alternatively you can upload the xform with one language, add the additional language, and then use Bulk Translation to add additional languages.
- Both tools described here uses the name of you Excel file as the root of all of the xpaths in your form. So if you call your file "survey" then your data will look like /survey/question4 instead of /data/question4. You cannot change this after uploading into CommCare. We recommend that you name your excel file "data" if you want to be consistent.
- The "add other option" for multiple choice questions will generate the "other" question with the correct question ID but will not generate the correct skip logic.
- You may also notice some extra "hidden values" at the bottom of your form after using the Excel tools. You should delete these. These hidden values are *formhub*, *uuid*, *meta* and *instanceID*.
- Advanced features link hint text, multimedia, read-only, and complex logic may not be well supported.

## Shortcut for creating the XLSForm format from CommCare's Export Form Contents

You may end up with an Excel form that is formatted like [Export Form Contents](#) as generated in CommCare, where all of the choices and questions are on one page. Below is a process to simplify breaking it into two pages.

The two parts of the process are:

- 1) split the choices from the questions (e.g. if the question is "What block do you live in" then the choices would be "Manjhanpur", "Mooratganj", "Kaneli", etc); and
- 2) format the sheets to match what XLSForm wants.

### Here are the steps

1. Create a new column in excel that is the name of the question for the choices. To do this, you can use an if statement. If you have the question names in column C and the type of question in column H, then it would be the following: =if(H3="Select Item",C3,C2) . This way, if you have a question, it will copy the question name, and if you have a choice, it will copy the value above it, which will be the question name from its question.
2. Create a new column in excel that equals 1 if the row is a choice and 0 if the row is a question.
3. Copy the two new columns that you created and paste special values them on top of themselves.
4. Sort the newly created column of 0's and 1's for question to split the questions and choices. Then copy the choices to their own sheet.
5. On the new choices sheet, get rid of the columns that you don't need anymore. Then change the column headings to match the format required.
6. Go back to the sheet with the list of questions. Use find and replace to change the names we have used for question type with what xforms uses (e.g. select\_one rather than single select, text rather than text question. So find and replace all "single select" with "select\_one").
7. Create a new column in excel. In this column, if the type of question is select\_one or select\_multiple, then the value should be "type nameofquestion". If it is any other type, then it should just be the type of question.

For example, if the type of question is select\_one and the name of question is consent, then the value in this column should be "select\_one consent". If it is a text question, then this value should be text. To do this, if the question name is in column B and the type of question is in column C, starting from the 3rd row, you can use =if(OR(C3="select\_one",C3="select\_multiple"), C3&" "&B3, B3).

8. Change the names of the columns to match the format for xforms and delete any that you don't want. You will need to delete the validations if you have not edited them for xforms format.

9. Upload the form following the instructions above