***Day 2 – Building Complex forms; ICT management***

**Day’s Objectives:**

By the end of the afternoon, meeting participants:

* Can build complex forms using iFormBuilder software
* Understand best practices of ICT device management

# Warm-up

**9:00 – 9:30**

**Facilitator: X**

**Session Objectives**

By the end of the session, participants:

* Will have had practice analyzing data in iFormBuilder

**Materials**

Equipment: LCD Projector,

Supplies: Flip Chart, markers

| **Time** | **Method** | **Facilitation notes: Introduction** |
| --- | --- | --- |
| 9:00 – 9:30 | *Warm-up exercise*  *(30 min)* | Using the data-set of the pre-ICT4E training questionnaire, instruct participants to answer the following questions as a warm-up exercise:   1. What percentage of workshop participants work for Caritas Germany? 2. How many males are participating in the workshop? 3. What is the average level of confidence in analyzing data collected with an ICT device? 4. On average, how confident are the female workshop participants with using surveys with forms developed in iFormBuilder?   Ask participants how they came up with their responses, with the key message being that there are different ways to analyze datasets using the iFormBuilder platform. |

# SESSION 1: INTRODUCTION TO DEFINITION FILES

# 9.30 – 11.00

**Facilitator: Sarah Gilbert**

**Session Objectives**

By the end of the session, participants:

* Have a basic understanding of what an ICT4D definition file is.
* Have had an opportunity to practice preparing a definition file for a single-form survey.

**Key messages**

1. Thought needs to be put into translating a form from paper-based to electronic formats. The ICT4D definition file is a tool that has been developed to help guide that conversation. Even if time is tight and the ICT4D definition file cannot be created, a conversation needs to take place between the M&E/program staff and the person creating the electronic form needs to happen to make sure the final product is correct.
2. In a traditional paper-based tool, guidance on skip logic is typically printed on the page for users’ information. In an ICT4D application, skip logic is not necessarily visible to the user. Rather, the program and M&E staff who design the content specify the required “show if” logic as part of defining the content which is then built into an ICT4D application. Once built, the application automatically shows questions that correspond with the logic specified by program and M&E staff in the content definition.
3. What you are designing a survey questionnaire, a monitoring tool or a mobile phone-based job aid, it is essential that every question/message in an ICT4D definition file has a unique Question Code. This not only ensures that you won’t have duplicate variables in the data generated from use of the application, but it also helps in developing accurate “show if” logic in the definition itself.
4. When designing content, staff can specify whether a response to a specific question is required or not. If required, the application will not advance until the user has entered a response. This helps to ensure data completeness.
5. **Constraints** (or validations) for individual questions can be added so that the application only accepts responses within an allowable range or that meet a pre-set criteria.
6. There is never only one way to define content, show if logic and question types in ICT4D applications. Staff should take into consideration ways to maximize ease of use, data quality, and data completeness when defining content for an ICT4D application.

**Materials**

Equipment: LCD Projector

Supplies: Flip Chart, markers, sticky notes

Handouts: 2.1.2 ICT4D\_definition\_File\_2.1.1;

| **Time** | **Method** | **Facilitation notes: Session 1 – Introduction to definition files** |
| --- | --- | --- |
|  | *Preparation* | * Print copies of Handouts for each participant |
| 9:30 – 10.30 | *Presentation*  *(30 min)* | Set the scene by showing Slide 11.  **ASK** participants how they, as neophyte ICT users, would convert paper-based forms to iForm.  **Key Message 1**: Thought needs to be put into translating a form from paper-based to electronic formats. The ICT4D definition file is a tool that has been developed to help guide that conversation. Even if time is tight and the ICT4D definition file cannot be created, a conversation needs to take place between the M&E/program staff and the person creating the electronic form needs to happen to make sure the final product is correct.  Refer to Handout 2.1.1 ICT4D\_definition\_File\_2.1.1  **Explain the 7 columns in the table**:   * “**Question Code**” is the same thing as the “Question ID” seen in the paper-based example. Whether you are designing a survey tool or a mobile job aid, it is essential that every question/row has a unique * Share **Key Message 2**: What you are designing a survey questionnaire, a monitoring tool or a mobile phone-based job aid, it is essential that every question/message in an ICT4D definition file has a unique Question Code. This not only ensures that you won’t have duplicate variables in the data generated from use of the application, but it also helps in developing accurate “show if” logic in the definition itself. * Unlike “skip logic” that is typically shown at the end (far right side) of a survey question telling you where to go next, an ICT4D application uses **“show if” logic** that appears before each question in the definition file. “Show if” logic defines when the ICT4D application should show each question or message in the definition file. * **Show if (skip logic) -** Description on when to show or hide this question with respect to data entered up to this point. You may refer to include questions previously asked in this form or on previous visits. * Share **Key Message 3:** In a traditional paper-based tool, guidance on skip logic is typically printed on the page for users’ information. In an ICT4D application, skip logic is not necessarily visible to the user. Rather, the program and M&E staff who design the content specify the required “show if” logic as part of defining the content which is then built into an ICT4D application. Once built, the application automatically shows questions that correspond with the logic specified by program and M&E staff in the content definition. * **Required?** Specify whether a response to a specific question is required or not. * Share **Key Message 4**: When designing content, staff can specify whether a response to a specific question is required or not. If required, the application will not advance until the user has entered a response. This helps to ensure data completeness. * **ASK**, “Can you think of a time when a response might not be required?” * Give participants a chance to respond. * If needed, share that ‘phone number’ is an example if not all of your beneficiaries have phones. You would want to be able to enter the phone number, if a beneficiary had one. But you wouldn’t want the application to refuse to advance to the next question if there was no phone number to enter. * **ASK**, “Can you think of a time when a response might be ‘not applicable’?” * Give participants a chance to respond. * Labels are an example where the “Required?” response is not applicable, since no data is entered for labels. * **Visible question text in English** – text as it will appear to the user   - **“Question type”** defines the type of question/response required for each question or message in the application. Examples of question types include:   * Labels require the user to enter no date. Rather, the label simply displays information to the user. Example, you can have a label that shows the beneficiary’s name at the top of each form. You can use a label at the end of a form to let the user know that the form is complete. You can use a label to convey a pre-defined counseling or behavior change message. * Single Answer questions are those that have a list of possible (multiple choice) options, but the application only allows one response to be selected. * Multiple Answer questions are those that have a list of possible (multiple choice) options from which the user is allowed to choose multiple responses. * Number (integer or decimal): You can specify in the application definition that only numbers are acceptable responses to certain questions. If you know, you can further specify that you want the application to only accept an integer (i.e., a whole number, not a fraction or decimal) or a number with a decimal point. * Text can be specified for questions that require that the response be recorded in narrative or text form (e.g., beneficiary’s name, etc.) * Date and/or Time can be specified as the type of response(s) to be entered for a given question. * Phone or ID number can also be specified as the type of response required/allowed for a question. * Other question types include *GPS* if GPS coordinates are required, *microphone or audio* if the response is being audio recorded, *camera or photo* if a picture is being taken, *barcode* if the response requires a barcode scan. * Explain that the definition file has a specific place where select options (multiple choice options) can be defined. Pre-defining and pre-coding multiple choice responses improves data quality and simplifies analysis. * **ASK**: “**How does pre-defining and pre-coding responses improve data quality? How does it simplify analysis?”** * **Share** the ***Design Tip*** about determining whether data analyst wants numeric or string variables: **During design, talk to the person who will do the analysis of data collected with the ICT4D application. Determine whether data needs to be available as numeric (number) or string (text) variables for the purpose of analysis. Then talk with the ICT team to make sure the application data is captured in the format required to best facilitate analysis.** * Share **Key Message 5**: **Constraints** (or validations) for individual questions can be added so that the application only accepts responses within an allowable range or that meet a pre-set criteria. * ***ASK*** if anyone can think of an example where data constraints could be included in the definition to improve data quality? * Allow time for response   If needed share examples: specifying a valid range of numbers for age, temperature, crop production, allowable date, etc.   * **ASK**  for and respond to any questions so far on definition files. |
|  | *Group Work 1 (30 min)* | * Divide participants into groups: * Distribute 2.1.2 Household registration form * **Explain** that groups have 45 minutes to prepare an ICT4D definition file based on the handout. For each question in the sample survey, the group should define:  1. Question Code 2. Show if logic 3. Required? 4. Question 5. Question type 6. Constraints  * Groups can either prepare flip charts showing their definition file or use the ICT4D Definition File to prepare an electronic definition file |
| 10:30-11:00 | *In Plenary*  *(30 min)* | * Distribute and display the Excel-based definition file for single-form survey v.1 * Allow groups time to review their own definitions (Content Planning Table) against the sample definition. * **ASK,** “Did any group define the survey content differently?” * Allow time for response. Let groups with other content definition share their content and the reason for choosing it. * Share **Key Message 6**: There is never only one way to define content, show if logic and question types in ICT4D applications. Staff should take into consideration ways to maximize ease of use, data quality, and data completeness when defining content for an ICT4D application. * **Ask** for and respond to any questions |
| ***COFFEE BREAK*** | | |

# SESSION 2: BUILDING COMPLEX FORMS

# 11:30 – 13.00

**Facilitator: X**

**Session Objectives**

By the end of the session, participants:

* Know how to create smart option lists
* Know how to set a default or fixed value for a given question.
* Understand skip logic for Multi-Select questions
* Know how to generate random ID numbers in iForm
* Know how to do calculations in iFormBuilder surveys
* Know how to create a warning message and an error message in iForm

**Key messages**

1. If you have a field that will always be a specific value and there is no need to have any other option (i.e. Distributions are always done in the same state) you can set the value of the text field using the “Dynamic Value” section of the Smart Control tab. Make sure that the value you would like to appear by default has quotes around it.
2. Option lists can be programmed to show different options – referred to as Smart Option Lists – depending on the answer to previous questions. This functionality is helpful when recording a program participant’s state/district/municipality of origin, which can be dependent on previous responses.
3. It is possible to create randomly generated beneficiary IDs and token numbers using iFormBuilder. These numbers can also include static text or references to registration locations, depending on how this information is collected. RISAs and ICT4D managers should be able to share more information about how to link randomly generated numbers to registration locations or similar data point.
4. Where possible, always consider using the calculation functionality in iFormBuilder since this is an easy way to keep data clean.
5. It’s good practice to create a warning to remind the data collector to double-check when calculations are done in iFormBuilder. These warnings can help data collectors verify and clean the data by using follow-up questions as the survey is carried out.
6. Error messages are a good way to help data collectors identify where survey responses do not make sense. Until the error is resolved, the data collector will not be able to completely save the survey.

| **Time** | **Method** | **Facilitation notes: Session 2 – Building Complex forms** |
| --- | --- | --- |
| 11:30 – 13:00 | *In plenary, each participant on their computer (90 min)* | Have participants begin to create this new form, based on their notes from the discussion session and the ICT4D\_definition\_file  *Note: remind participants to name their forms and option lists uniquely by including their username in the titles.*  STEP 1:  **ASK: Who can explain to the group how to create a new form based on what was learned earlier in the training?**  **ANSWER:**   * Go to the Form Builder section of the Forms tab * Select “Create New Form” * Have participants choose a unique title for their form, as well as a new icon. * Save the new form   *Note: The next step can be skipped if the group is not as advanced or would not like to format their forms.*  STEP 2:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the first field be?”  **ANSWER**: In the case of our example, the first field would be a Label field with the purely informative text “A. Distribution Site Information”   * All participants should now have a form with one label (go around and check each computer to make sure everyone is keeping up. Do this continually throughout the exercise). * Make sure that participants notice that making a field a “Label” makes it read-only by default.   STEP 3:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next field be?”  **ANSWER**: The next field would be a field where the State where the Registration is taking place can be captured. In our case, we want the value of the text field to be pre-set to “Lakes State” (or another CP appropriate value)   * Review the **Smart Control** tab of the Element Properties. Fix the value of the text field using the “Dynamic Value” section of the Smart Control.   SHARE **Key Message 1:**  **If you have a field that will always be a specific value and there is no need to have any other option (i.e. Distributions are always done in the same state) you can set the value of the text field using the “Dynamic Value” section of the Smart Control tab. Make sure that the value you would like to have appear by default has quotes around it.**  STEP 4:   * Have participants repeat the Dynamic Control default value for the next two elements of the form “County” and “Municipality”. * Have them also create the Text field “District” * Then have them sync the devices and test. * Support the participants as needed.   *Note: The next step can be skipped if the group is not as advanced or would not like to format their forms.*  STEP 5:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next fields be?”  **ANSWER**: In the case of our example, the form now begins to ask data that is specific to the respondent instead of the Registration Location, so the next field would be a Divider field (note that the label and column name for the divider are rather unimportant, since no data will be displayed or captured here). The Divider field would be followed by a Label field with the purely informative text “A. Registration Information”  *Note: Participants should be comfortable enough now with interpreting the information on the sheet that they can begin to work individually, with the instructor helping as necessary.*  STEP 6:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next field be?”  **ANSWER**: The next field would be the respondent’s name, as a text field.  STEP 7:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next field be?”  **ANSWER**: The next field is a required Select input type, yes/no question “Is the respondent the Head of the Household?” The Option List for Yes/No should already be created in the Profile, so remind the participants that they should use that generic Option List.  STEP 8:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next field be?”  **ANSWER**: This question will need to use skip logic, since it will only appear when the respondent is not the head of household.   * Review the **Smart Control** tab of the Element Properties. Review the Condition Value functionality – where the column name of the previous question should appear (is\_the\_respondent\_the\_head\_of\_the\_household==1)   STEP 9:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next field be?”  **ANSWER**: The next field is a required Select input type, yes/no question “Are you an IDP or Host family?” The Option List for Yes/No should already be created in the Profile, so remind the participants that they should use that generic Option List.  STEP 10:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next fields be?”  **ANSWER**: The next three questions will need to use skip logic, since it will only appear when the respondent is an IDP.   * Review the **Smart Control** tab of the Element Properties. Review the Condition Value functionality – where the column name of the previous question should appear (are\_you\_an\_idps\_or\_host==0) * Create a Smart Option List so that the question on the IDP’s County of Origin can be filtered to only present counties that are in the State selected (potential responses are shown in ICT4D\_Definition\_File). In the Advanced Option List tab, use Javascript logic to show when different options should appear (i.e. if\_idp\_which\_state\_do\_they\_come\_from==0) Be sure to group the answers based on their dependent answers (e.g. if answer to first question is State A – all municipalities in State A should be listed together in the option list answers). The condition only needs to be included for the first answer that should appear. The program logic will show all the answers following the condition until a new condition is created. * Make sure that the “Other” options have their associated text fields, using skip logic so they only appear when “Other” is chosen.   SHARE **Key Message 2:**  **Option lists can be programmed to show different options – referred to as Smart Option Lists – depending on the answer to previous questions. This functionality is helpful when recording a program participants state/district/municipality of origin, which can be dependent on previous responses.**  STEP 11:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next fields be?”  **ANSWER**: The field should be Read-Only, as it should be a randomly generated number to be created by iFormBuilder. Random number generation is a two stage process:  *Stage 1:* Go to the Title page of the form where you need the ID generated (use the “Edit” button next to the name of the form when in iFormBuilder).  At the very bottom of that page, put the following text in the “page level javascript” area:  var genguid = function b(a){return a?(a^Math.random()\*16>>a/4).toString(16):([1e**6**]+'').replace(/[018]/g,b)}  *Please note: The number in bold tells you how long your random alpha-numeric ID will be – the length of the random ID will be that number +1 (in this case, it will be 7).*    *Stage 2*: The field where the random number needs to be generated should be a Read-Only text input, so no modifications can be made.  In the **Smart Page**, put genguid() in the Dynamic Value field (no quotes necessary).  If you want to put text before the number, put “CRS-“ + genguid() and the text “CRS-“ will appear before the number (Of course, you can put any text, it does not have to be CRS). The same goes for text after the number as well.  SHARE **Key Message 3:**  **It is possible to create randomly generated beneficiary IDs and token numbers using iFormBuilder. These numbers can also include static text or references to registration locations, depending on how this information is collected. RISAs and ICT4D managers should be able to share more information about how to link randomly generated numbers to registration locations or similar data point.**  STEP 12:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next fields be?”  **ANSWER**: The next four fields are required Number fields where the number of male and female adults and children in the household are captured.  STEP 13:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next fields be?”  **ANSWER**: The total number of household members can be calculated by adding the answers to the number of female and male adults and children that were asked previously. Calculations can be done in iFormBuilder by summing the data column names of the fields in question in the Dynamic Value element of the Smart Control option (e.g. total\_no\_of\_household\_members=no\_of\_male\_children\_under\_18+ no\_of\_female\_children\_under\_18+ no\_of\_males \_over\_18+ no\_of\_females \_over\_18). Since the Total Household No. field is “Read-only” the number will not be able to be modified without adjusting the household numbers previously entered.  SHARE **Key Message 4:**  **Where possible, always consider using the calculation functionality in iFormBuilder since this is an easy way to keep data clean.**  Since the total household number is calculated automatically, it is good practice to create a warning that appears only when the calculated field is greater than 0 that asks the data collector to verify the total household number. Warnings can be created by making a Label field (in our example case, containing the text “\*\*Please ask respondent how many total household members they have to verify this number.\*\*”) that appears only when the field for the calculated family members is greater than 0, by putting in the proper JavaScript condition in the Condition Value portion of the Smart Control tab.  SHARE **Key Message 5:**  **It’s good practice to create a warning to remind the data collector to double-check when calculations are done in iFormBuilder. These warnings can help data collectors verify and clean the data by using follow-up questions as the survey is carried out.**  An error message can also be created to ensure cleaner data – making sure that the total family members do not go over 20, for example. In a country with low family sizes, it could be worthwhile to create an error message when family size is greater than 20, for example, since the answer indicates that data was input incorrectly. Error messages can be created in two steps, both in the Smart Control section. The CLIENT VALIDATION is the area where the validation condition is entered - the client validation value must be true for a particular element to pass validation. The VALIDATION MESSAGE must be in quotes and supplies the error message text that will appear, if the client validation condition is not met.  In our case, the Client Validation should be total\_no\_household\_members <=20 and the Validation Message would be “Family size cannot be greater than 20.”    SHARE **Key Message 6:**  **Error messages are a good way to help data collectors identify where survey responses do not make sense. Until the error is resolved, the data collector will not be able to completely save the survey.**  STEP 14:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next fields be?”  **ANSWER**: The next field is a Multi-Select which lists all the possible household members who could collect water – list provided in the ICT4D\_Definition\_File. “Other” is an option and we should create a text field where the family member type can be entered when “Other is selected”.   * In the Condition Value field of the Smart Option section of the “If other family member, please specify” question, the condition should be:   ((data\_column\_name & 1) == 1) where the “data\_column\_name” is the name of the Multi-Select field for which family members collect water.  STEP 15:  **ASK**: “Based on the ICT4D\_Definition\_File and the discussion we had about the form design, what input type would the next fields be?”  **ANSWER**: The final questions help identify the vulnerability status of the households. To answer these criteria, data could either be captured in a multi-select or as separate questions (i.e. Is there a pregnant mother in the HH? Y/N; Is there an elderly person over 65 in the HH? Y/N;…) In the case of the Multi-select option, there is no way to guarantee that the data collector really asked all the vulnerability questions, so the best practice is to have all the vulnerability questions separated out as Y/N questions in iForm.  STEP 16:   * Save, sync and test! |

**LUNCH BREAK**

# SESSION 3: Device Management – Good practices

**14:00 – 14:30**

**Facilitator: X**

**Session Objectives**

By the end of the session, participants:

* Are aware of existing tools to support proper device management when integrating ICT into emergency programming

**Key messages**

1. Proper device set-up, inventory and management is half the battle!

**Materials**

Equipment: LCD Projector

| **Time** | **Method** | **Facilitation notes: SESSION 3: Device Management – Good practices** |
| --- | --- | --- |
| 14:00 – 14:30 | *Plenary* | Direct participants to the “Device Management” section of the Dropbox folder and review the contents:   * Android and iOS checklists for devices * ICT set-up documentation * Checklist of what to bring when deploying ICT in the field   SHARE **Key Message #1 –** proper device set-up, inventory and management is half the battle. |

# SESSION 4: “ICT in the Field” – Lessons learned with ICT4E

**14:30 – 15.30**

**Facilitator: X**

**Session Objectives**

By the end of the session, participants:

* Are familiar with lessons learned from CRS’ implementation of ICT around the globe.

**Materials**

Equipment: LCD Projector

| **Time** | **Method** | **Facilitation notes: SESSION 4: “ICT in the Field” – Lessons learned with ICT4E** |
| --- | --- | --- |
| 14:30 – 15:30 | *Summary (30 min)* | Walk through the summary of Lessons Learned with the participants.  **ASK:** How will participants incorporate these lessons learned in their current programming? |

**COFFEE BREAK**

# SESSION 5: Form Building Practice

**16:00 – 17:15**

**Facilitator: X**

**Session Objectives**

By the end of the session, participants:

* Are familiar with lessons learned from CRS’ implementation of ICT around the globe.

**Materials**

Equipment: LCD Projector

Handouts:

| **Time** | **Method** | **Facilitation notes: SESSION 5: Form Building Practice** |
| --- | --- | --- |
| 16:00 – 17:00 | *Form Building practice* | In small groups or individually, have participants practice building forms in iForm. If necessary, they can pick from the forms provided in the Dropbox (2.5.1 - KAP WASH single form survey, 2.5.2 - Monitoring of Post Distribution, 2.5.3 - Monitoring template (Assam Floods 2012)) |
| 17:00 – 17:15 | *Flip Chart and sticky notes* | Before people leave, draw a line down the middle of a flipchart paper with a happy face on one side and a sad face on the other. Hand out sticky notes and have people write what they liked and disliked about the day on the notes. Have them post the notes on the appropriate side of the flip chart as they leave. |