

Tangerine®

Full User's Guide

v. April 2017



Authors:

Scott Kipp, RTI International skipp.Contractor@rti.org

Lachezar Hristov, RTI International lhristov@rti.org

Carmen Strigel, RTI International cstrigel@rti.org

Sarah Pouezevara, RTI International spouez@rti.org

About Tangerine®:

Tangerine is open-source electronic data collection software designed for use on Android mobile devices. Its primary use is to enable recording of students' responses in oral early grade reading and mathematics skills assessments. Tangerine is also used to capture interview responses from students, teachers, and principals.



This work is licensed under a Creative Commons Attribution-ShareAlike 4.0 International License.

Note: This user's guide assumes some knowledge of EGRA/EGMA/SSME survey methodology and is not a methodological guide to conducting these types of oral assessments and surveys. For further information on administering the content of the assessments themselves, please refer to the toolkits at www.eddataglobal.org.

Contents

Overview	2
Tangerine Icon Library	2
A. The Tangerine Wizard	3
1. Logging in and Entering a Group	4
2. The Tangerine Group Homepage	4
2.1 Changing your Group's Language	5
3. Importing an Assessment	5
4. Creating a New Assessment	6
5. Building an Assessment	7
6. Editing Individual Subtests (Subtest Editor)	8
6.1 The Date and Time Subtest	10
6.2 The Location Subtest	11
6.3 The Student ID Subtest	11
6.4 The Consent Subtest	11
6.5 Grid pages or Timed Subtests	12
6.6 Survey pages (Untimed and Question and Answer Subtests)	15
6.7 Validation and Skip Logic	21
6.8 Subtest Skip Logic and 'Action on Display'	24
6.9 Randomizing Subtests	26
7. Duplicating and Deleting Assessments	27
8. Accessing, Downloading and Interpreting Data	27
8.1 Interpreting Data	28
8.2 Default and Automatic Variables	28
8.3 Converting Epoch/Unix timestamps	29
8.4 Data from Instruments with Non-Latin Characters	30
9. The Tangerine Results Dashboard	30
B. Tangerine on Mobile Devices – the “app”	31
1. Hardware Selection and setup of Tangerine software	31
2. Installing Tangerine on your tablet	33
3. Logging into Tangerine on mobile devices	34
4. Instruments on Tablets	34
5. Training and Data Collection	35
6. Entering data on tablets	37
7. The Student/Participant Identifier	40
Annex 1: Training Topics and Example Agenda	42
Annex 2: Tangerine Tipsheet	42
Annex 3: Guide to Supervisor Responsibilities during Data Collection	43
Annex 4: Tips for keeping Tablets Charged	44
Annex 5. Additional Examples of Skip Logic Commands	47

Overview

Whether you are using Tangerine for educational assessments, surveys, or both, the process for using the software involves the configuration of your instruments in Tangerine’s online environment (the “Wizard”) and thereafter using your instruments on an Android device (the “app”). This manual is structured in the two components making up the Tangerine Platform:

- 1.) The Tangerine Wizard, and
- 2.) The Tangerine application (app) on mobile data collection devices.

The Tangerine Wizard, is the web-based environment where data collection instruments are managed, that is developed, edited, tested, and deleted; and where assessment data is accessed and exported.

The Tangerine application is the device-level environment in which data is being collected, and from which data is being sent to the server (and then accessed via the web-based environment).

RTI International is proud to offer Tangerine’s software code as an open source piece of software under a GNU General Public License. To access the software’s code base, please visit our Github page. While we will make every effort to ensure a productive and seamless Tangerine experience for you, we cannot guarantee that the software will function perfectly at all times. Please be sure to check your Tangerine instruments and data files regularly and report any issues to RTI International so that we may assist.



*A note on browsers: the **Tangerine Wizard*** at this time works best in Google Chrome or Mozilla Firefox, both available as free web downloads. The Wizard will not function in Internet Explorer, and may experience problems in other browsers. The **Tangerine Application** on Android tablets is actually running through the device’s browser, so be sure not to delete or disable your device’s browser. If your device can be equipped with Google Chrome, we recommend this.*



Tangerine Icon Library



Tangerine icon. In the Wizard and on the Tangerine App, tapping this icon will bring the user back to the main landing page. If this icon is spinning, this indicates that the application or webpage is performing.

Groups

This button will return the user to the main landing screen, where all the user’s groups are visible.

APK

This button will produce an Android Package (.apk) file, which contains Tangerine and all of the active

instruments in a group. This will generate a link to download your group's application file to install onto Android devices.

Results

This displays the Results Dashboard, not to be confused with the raw data itself (see graph icon below).



The 'Run' button launches an assessment, whether being used to collect data or simulate an instrument for quality control testing.



The 'Data Entry' mode for an instrument can be used to enter results through the online Wizard.



The 'Data' button. In the **Wizard**, this will bring the user to the results download. On **tablets**, this button will display the register of complete and incomplete assessments.



The 'Edit' button is used to enter into an assessment for modification and/or adding new subtests.



The 'Sync' button on the **Wizard** is used to synchronize instruments built offline with instruments built online (see 'Tangerine Satellite' section). On **tablets**, this button is used to update instruments, synchronizing tablet versions with edits from online.



The 'Print' button in the Wizard displays options for viewing the content of all subtests in an assessment.



The 'Duplicate' button in the Wizard will create a new copy of an assessment. Doing this will clear away any trace of removed variables from your instrument's data file.



The 'Copy to' button in the Wizard can be used to copy questions between subtests and subtests between assessments.



The 'Delete' key will remove an assessment from the **Wizard**.

A. The Tangerine Wizard

To begin building instruments in Tangerine, you must establish a user name and password on the Tangerine server you are using. Tangerine can be used by establishing a server hosted by RTI International, or by as hosted by anyone who has installed Tangerine's open code on their webhost. Please see www.tangerinecentral.org to learn more.

Both your username and password are case-sensitive. Once logged in, your user name will appear in the top right-hand corner of the browser window.

1. Logging in and Entering a Group

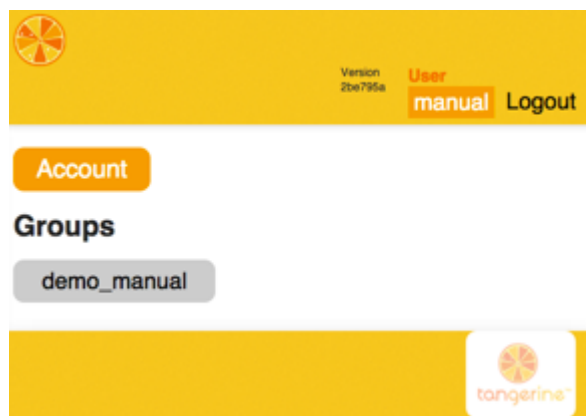
Instruments and accompanying data in Tangerine are organized by ‘Groups.’ After you have chosen a user name and password, you will see the landing screen indicating that your user is not part of a ‘Group.’ Once you have logged into Tangerine, you can either:

1. Join an existing group; or
2. Create a new Group.

If you would like to join an existing group, please contact the person who has created the group you would like to join and request that your user name be added to that group.

If you would like to create a new group, you must login with username as: ‘user1’ - this is the only username on a Tangerine server with the ability to create a new group. If you do not have the password for this account, please contact your server’s administrator to request that a new group be created.

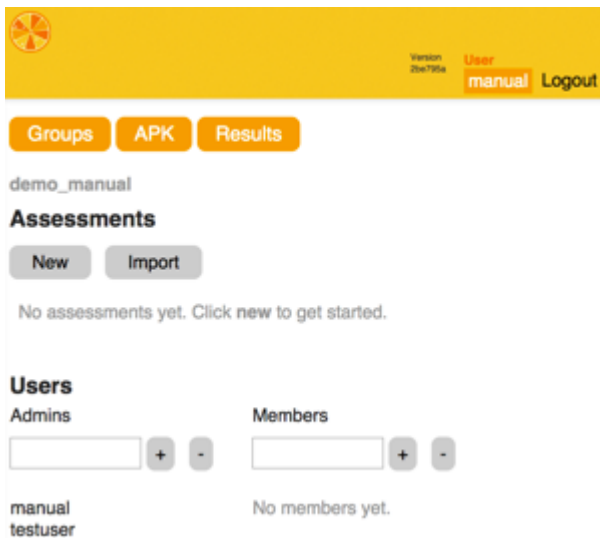
Once logged in as ‘user1’, click the “Account” button shown in the image above and then select “Join or create group” on the page that follows. You will be asked to re-enter the ‘user1’ password in order to create a new group. Once you have done this, you should now see your new group appear in the Wizard landing screen. In this example, the group “demo_manual” now appears as an option.



Try to create group names that are unique to your work and thus identifiable. Consider including the name of your organization and/or the country and date where your instruments will be used. You may also wish to create a “Sandbox” group where you experiment with instrument building.

2. The Tangerine Group Homepage

Once you have joined or have been given access at least one group, you will see buttons representing each group. Click a group button to enter into that group to work in it. Once you are inside a group, your screen (as in the example below) will present several options:



In Tangerine, all instruments are referred to as ***Assessments***. You can:

1. Create a new assessment (see below).
2. Import an assessment from another group (see below).
3. Add collaborating users.

You can add collaborating users by entering their account names into the text fields under ‘***Admins***’ and ‘***Members***.’ Admin users have all privileges for editing, deleting and duplicating instruments. Member users, also known as ‘readers’, can only simulate (run) assessments, download their results files (.csv files), and print out instrument contents. To add a user into either role, enter the user’s name in the text field and click the ‘+’ button. Enter your password to confirm.

Recall that user names are case-sensitive. If you see an error when trying to add a user to your group, confirm the exact spelling of the username with the user, and ask them to confirm that they have logged in on the Tangerine server you are using.

2.1 Changing your Group’s Language

English is the default language of both the Tangerine Wizard and the Tangerine app that is produced by the Wizard. At this time it is also possible to change your group’s language to French. We hope to provide additional languages in the future.

To change your group’s interface language:

1. Login to your group.
2. Hover your mouse over your username (in the top right-hand corner of your browser).
3. Select ‘Settings’.
4. In the field titled, ‘Language’, change the value from ‘en’ to ‘fr’ to have the group in French. Click ‘Save’ at the bottom of the ‘Settings’ page.

3. Importing an Assessment

If you have instruments developed in another group which you would like to repurpose in a new group, you can import them by clicking the ‘Import’ button found at the top of your group’s main page:

1. You must be an Admin of the group you are importing from;
2. In the original group, copy the desired instrument's **download key**;
3. In the target group (where you want the new copy), click 'Import'.
4. Select the original group and enter the download key in the box. Click 'Import'.
5. Return to the new group's main page and you will see the import.

Note that the new copy of the imported instrument does not include any results data that may have been uploaded to the original copy. Any results uploaded to any instrument will only appear in the group from which the copy of the instrument came. If you try to import an instrument that is archived in the original group, it will appear as archived also in the new group.

4. Creating a New Assessment

In Tangerine, assessments are comprised of **subtests**. Each subtest in the Wizard will become its own screen on the tablet application. This section will walk you through the process of setting up a new assessment with multiple subtests.

1. Click the "New" button just below the 'Groups' button on the Group homepage and a box will appear for you to name your new assessment, as shown below:
2. Name your assessment, then click "Save." Your assessment will now appear in the "Assessments" section of the group's landing page. Assessments on this page are categorized alphabetically.
3. Once named, you can edit your assessment by clicking on the orange arrow to the left of its title. Options will then drop down from there, as shown below:

Assessments

New

Import

▼ Test



Download key b3b71

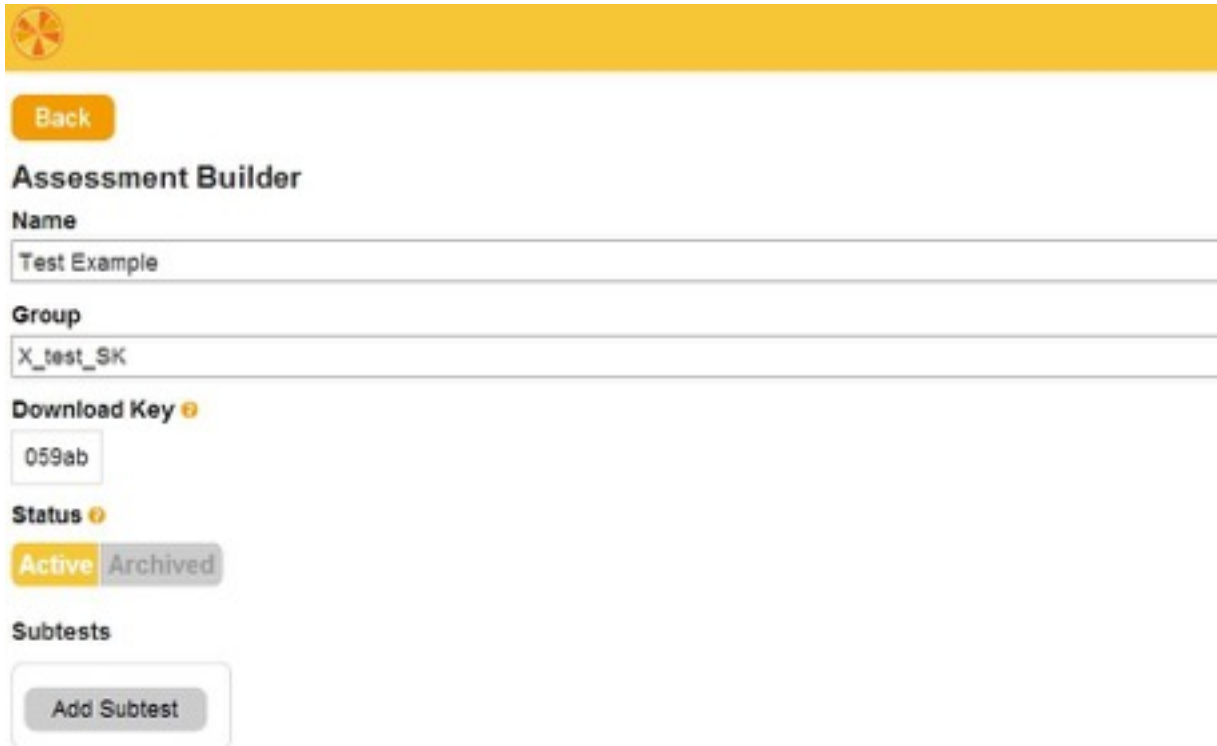


Carefully select the name for your assessment. Make sure to include the name of the project or country, month of data collection, assessment target group and potentially some other unique details. If you expect that you will be using Tangerine several times in a specific project (e.g. for baseline, midterm and final data collection or annual data collection activities), it may be helpful to include dates in the assessment name. You will want to make sure that you are using appropriate instruments and that these are clearly identified.

5. Building an Assessment



To start building your assessment, click on the edit icon () and you will arrive at the Assessment Builder page, as shown below:



The Assessment Builder interface features a yellow header bar with a Tangerine Wizard logo on the left. Below the header is a 'Back' button. The main title 'Assessment Builder' is followed by a 'Name' label and a text input field containing 'Test Example'. Below this is a 'Group' label and a text input field containing 'X_test_SK'. The 'Download Key' label is followed by a text input field containing '059ab'. The 'Status' label is followed by two buttons: 'Active' (highlighted in yellow) and 'Archived' (greyed out). At the bottom, the 'Subtests' label is followed by a button labeled 'Add Subtest'.

Change “Name” by typing into this field. The download key and ‘Group’ are not changeable. The ‘*Status*’ of an assessment will be highlighted in yellow as either “Active” or “Archived” – which you can change by clicking on the button (when the button is yellow, it is selected). This setting only affects whether an assessment is included into an APK file; when you export your APK file from the Tangerine Wizard, only ‘Active’ assessments will be included. An assessment’s status has no effect on data – both *Active* and *Archived* instruments can receive data being uploaded from tablets. Your ability to edit subtests is also unaffected by the assessment’s status.

Click ‘Add Subtest’ to create a new subtest, which will generate the ‘Name’ prompt and ‘Type’ menu:

Please select a subtest type

- consent
 - Consent Page
- datetime
 - Date and Time
- survey
 - Student Information
 - Rational Counting
 - Missing Number
 - Number Discrimination
 - Word Problems
 - Pupil Context Interview
 - Listening Comprehension**
 - Initial Sounds
 - Untimed Reading Comprehension
 - Addition Level 2
 - Dictation
 - Phoneme Segmentation
 - Timed Reading Comprehension
- location
 - Consent Page

Subtests

New Subtest

Type

Please select a subtest type

Name

Add **Cancel**

Subtest Type and Name: See the sections below for distinctions on the various subtest types. The “Name” is what will appear in the application when you send it to a device, and as such can be where you enter the local language title for each subtest.

Usually, each assessment begins with four standard subtests capturing basic data:

1. The date and time subtest;
2. The location subtest, used to select your point of data collection from a list (e.g. select a school name from a filtered list of schools);
3. The student id subtest, used to create a unique, anonymous data identifier; and
4. The consent subtest.

Additional subtests will be made up of “survey” or “grid” pages, the page formats most often chosen for the various EGRA/EGMA assessments and surveys. Subtests will appear in the order in which they were added. To reorder your subtests, simply click on the left-hand icon of three gray bars to drag pages/subtests into a different order:



Click and drag the handlebars to reorder subtests and questions.


6. Editing Individual Subtests (Subtest Editor)


Click on the edit (pencil) icon next to the subtest you want to edit, as shown below:





To the right of each subtest, you will notice four things:

1. Type of subtest: this is the system-generated name of the prototype you selected for each subtest (e.g. 'grid' from the above screenshot).

2. The edit button, , which you will click to edit subtest content.

3. The “Copy to...” button, , which you can use to send a copy of a given subtest to any other assessment in your group.

4. The delete button, , which can be used to remove subtests from your assessment. You will be prompted to confirm this choice. NB: Deleted assessments cannot be recovered. If you think you may need an assessment or its data at a later stage, it is best to set that instrument’s status to “Archived” rather than deleting it.

After clicking the edit icon, , you will land on the Subtest Editor page. At the top of this page, you can edit the name of your pages/subtest as needed. Below the field showing the name of your subtest you will note the subtest’s “Prototype” – this is system information on the type of page template you have chosen; this cannot be edited.

Subtest Editor Terms and Settings:

The following fields and configurations appear in all types of subtests. Other settings and options are unique to certain types of subtests (such as grid subtests, survey subtests).

Language Code: here you can indicate the language used for the subtest’s content. This is a reference field only and will not affect how your content is displayed in Tangerine.

Skippable: you can indicate whether this subtest is required or not. The default setting of ‘No’ means that all items will be required before a user can advance to the next subtest. Requiring your users to complete a subtest will avoid missing values in your dataset (provided individual questions are also kept with ‘Skippable’ set to ‘No’). When set to ‘Yes’, a button labeled ‘Skip’ will appear on the bottom left of the tablet screen.

Display Back Button: you can allow a ‘Back’ button to appear. Enabling this will allow users to navigate back and forth between subtests. When left at the default ‘No’, users should be instructed to only press ‘Next’ when they have confirmed that all data entered is accurate.

Enumerator help: you can include a description of the subtest assessment procedure for the enumerator’s benefit, or any other instruction that may not always need to be visible. This information can be collapsed/expanded by the enumerator with one touch on the “Help” button during data collection to avoid taking up screen space on devices.

This ensures that the information is easily accessible in case there is doubt as to how to administer the subtest and can provide helpful reminders.

Student Dialogue: typically used to display the dialog that your data collector says to the student/participant (i.e. the actual instructions they read out loud). The text editor is used to bold, italicize, underline, or accent in any way a piece of information. Usually, the words that are to be read out directly to the participants are **bolded**, while instructions to the enumerator are in normal typeface.

Transition Comment: here you can add text that will appear at the bottom of the subtest screen in Tangerine, just above the “Next” button that advances the user to the next subtest screen. It can be used to enter reminders or further instruction, such as “Take the reading passage away from the student before proceeding” or “Remember to confirm that all information on this screen is correct before pressing ‘Next.’”

Preferred Font: here you can instruct Tangerine to apply (embed) a Unicode UTF-8 font of your choosing. This is important for some languages and scripts whose characters may not be contained within the character libraries of the tablet you are using. At present, embedded font options include: andika, gentium, padauk, zwekabin, rabiati, and Mondulkiri.

If you are having trouble displaying some characters or fonts on your tablet and you need support for additional font libraries to be embeddable options, please contact RTI or your Tangerine host provider.

Note that though some characters may display properly when viewed on a computer’s browser, an Android tablet may not contain the same character library, and so the tablet may hide or distort some characters – always verify your instruments’ fidelity of display on your tablets.



When pasting text into the Enumerator Help, Student Dialog or Transition comment boxes, first click on the icon marked “Paste as plain text” in the top row of the editor, the fifth icon from the left (see above screenshot for Student Dialog). Paste the text into the box that appears, click “OK”, and the text will appear in the Tangerine Wizard. If you copy text directly from a rich text program such as Microsoft Word, your copied text will likely be accompanied by additional formatting which may distort the text in the kinds of wizards (aka “WYSIWYG’s”) used by Tangerine. This additional markup formatting may cause other errors in your instruments; if you begin to see odd formatting or text displaying incorrectly on screens, check that your dialog texts do not contain additional formatting by copying all text into a plain text editor (such as Notepad or TextEdit), deleting it from the dialog box in Tangerine, and re-copying back in from the plain text editor.

6.1 The Date and Time Subtest

This subtest is used to generate date and time variables in your data set. When this subtest appears on the tablet screen, it will auto-populate the date and time variables with the current settings from the tablet. Typically, data collectors should just press the “Next” button if the data should reflect the time that the data collection actually occurred. If your data collectors notice that the date and/or time displayed in Tangerine on their tablet is not accurate, then this is an indication that the tablet’s date and time settings are incorrect. In these cases, the user should change the date and time in the tablet’s settings and restart the assessment on the tablet to have Tangerine reflect the correct time.

The Date and Time subtest’s functions cannot be edited. If desired, data collectors can manually adjust the date and time which appear on the screen (e.g. such as when entering into Tangerine data which was originally collected on paper at an earlier time/date). Editing the displayed date and time on the Tangerine tablet screen will not change the tablet’s date and time settings, it will only change the date and time data associated with that particular assessment.



The Tangerine app will pull the date and time information from the device on which it is installed. To

ensure accurate time stamps for your data, verify that the date and time is configured properly and consistently on all of your devices prior to data collection.

6.2 The Location Subtest

The Location page allows assessment creators to customize location labels (variables) for the country/region of data collection. You can create multiple fields for specifying your data location sites. These are what you enter into the field labeled, ‘Geographic Levels.’ In the example below, the labels are ‘Province’, ‘District’, ‘Name’ and ‘ID.’ The default option for separating these fields is by the use of a comma, as indicated by the highlighted “Comma” button. Each of these geographic level labels will be its own field for your data collectors to enter site-specific location info:

The image shows two side-by-side screenshots of the software interface. The left screenshot is titled 'Location data' and shows a text input field containing two lines of text: 'Central,Lilongwe,Lilongwe,1142' and 'Blantyre,Blantyre,Blantyre,2142'. Below the input field is a 'Format' section with two buttons: 'Tabs' and 'Commas', with 'Commas' being highlighted in yellow. The right screenshot is titled 'Geographic Levels' and shows a text input field containing the text 'Province, District, Name, ID'. Below this is a 'Format' section with 'Tabs' and 'Commas' buttons, with 'Commas' highlighted in yellow.

In the field for “Location Data” you can enter the site location details for the collection points in your sample. Once you have loaded the Geographic Levels and respective data, the data collectors filling out this subtest in the field will be prompted to select site location details from the list you have entered in this subtest.

If you receive this error, >“Some columns in the location data do not match the number of columns in the geographic levels.”

This is an indication that your location data may be incomplete, or you may have pasted in extra commas somewhere in your Location Data. You may find it easiest to open your data in a spreadsheet program and search for commas in the names of schools, replacing them with another symbol such as a hyphen or a slash.

6.3 The Student ID Subtest

The Student ID subtest is used to generate a unique ID code for your data record. This subtest’s function cannot be edited. Although there is provision for Enumerator help and instructions, these are usually not needed. Data collectors simply press the ‘Generate’ button as shown below, then ‘Next’ to proceed, once a code has appeared in the box below ‘Random Identifier.’



If you intend to track specific survey participants across multiple assessments, multiple enumerators or over a span of time, it will be important for the enumerator or the participant to retain their Tangerine-generated ID number. You cannot assign numbers ahead of time to participants, such as “ABCDEF”, as the Tangerine system follows an algorithmic pattern to generate ID numbers. You have two options in using this type of subtest: either click “Generate” to execute a new ID number, or enter an ID number that has been previously created by the Tangerine system. If you want to require users to enter an ID number that has been previously generated by Tangerine, consider hiding the ‘Generate’ button (see appendix for applicable command).

6.4 The Consent Subtest

The consent subtest is used to show in your data that the assessment participant or survey responder agreed to proceed. This subtest generates a “Yes / No” question. If “Yes” is marked as the answer, the instrument will proceed. If “No” is marked, this indicates non-consent, and the instrument will automatically skip to the end of the instrument.

At the bottom of the Subtest Editor page for Consent, you will find a box titled “Consent prompt” - this is where you can customize the phrasing and language of the consent prompt to the specific assessment target group (e.g., “Does Student consent?”, or “Does the Principal consent to participate in the interview?”).

As you will see with other subtests, a similar process of filling out a “prompt” will occur. A “prompt” in Tangerine indicates the text that will be seen by the data collector asking the questions or collecting information from participants. Almost all of the other subtests are either of the “survey” format (question and answer, or untimed assessment formats) or “grid” format (timed subtests). Only one “Consent” subtest type will function properly in each instrument. For options on additional consent cut-offs, see section 6.8.

6.5 Grid pages or Timed Subtests



Grid subtests are most commonly used for timed subtests, such as Oral Reading or Letter Name Recognition. In addition to the standard fields for Enumerator Help, Student Dialog and Transition Comment, there are fields and settings unique to these timed tests that will appear in the Subtest Editor for any grid subtest prototype.

The **Variable name** you choose, such as “letters”, will be the prefix added to each grid item in your results file. In this case, each grid item (each letter) in your results file will be listed as “letter_1”, “letter_2”, etc., based on their sequential position in the **Grid items** box.

In the field for “Grid Items”, simply paste in the items (letters, words, numbers, etc) you have selected for this subtest. Delimit each item with a single space (if you have extra space breaks the system will remove these); Tangerine will generate grid buttons based on this spacing. For example, if you want the equation “5+3=(8)” to appear on one grid button, enter the equation with no spaces between its characters, adding spaces in between equations to indicate a new button on the grid. After entering your variable name and grid items, you can alter several configuration settings for how your grid will look and behave:

Right-to-Left direction: This setting will reverse the default order of scoring for grid tests.

Randomize Items: will shuffle the order of your grid items for each assessment case.

Layout mode: ‘Fixed’ will stabilize the grid’s settings irrespective of the tablet’s screen orientation, ‘Variable’ will attempt to adjust to screen optimization. ‘Fixed’ is the default and recommended setting.

Grid font size: Medium is the default. Select “small” if you have particularly wide grids (e.g. a grid of vocabulary words that are particularly lengthy). When testing on tablets, it is important that your users have enough blank space in the margins next to the grid to scroll without inadvertently tapping assessment items.

Capture item at specified number of seconds: This option allows you to take a ‘snapshot’ of a student’s progress at a specific time point.

Capture last item attempted: Ensures that your enumerators will be forced to indicate the end point of the student’s progress through the grid items.

Mark entire line button: For all but the first line of grid items, if this setting is left at ‘Yes’, a button with an asterisk “*” will appear to the right of each grid row, allowing assessors to quickly mark incorrect an entire line of

items. This button can be helpful in cases where a student has skipped over an entire line of items, as it facilitates quickly marking the entire line incorrect, allowing the assessor to maintain focus on the remainder of the subtest.

Columns: Indicate in how many columns you want to display the items (for letters we usually select 10 columns of 10 items each, for familiar words 5 columns of 10 items each). If your grid is displaying too wide on your device, consider reducing the number of columns.

Autostop: the number of consecutive incorrect items, counting from the first grid item, after which the subtest stops automatically. The autostop calculation will only activate if the first grid item is marked incorrect. For example, with an autostop value of 10, if a child has the first 4 items correct and then the following 10 items incorrect, the test will not autostop. It will only autostop if the **first** 10 items are all incorrect.

Timer: The total number of seconds you would like to allow for grid items to be attempted. If you would like to use the grid test setup, but not the timer, set the Timer value to zero and the “Capture last item attempted” setting to “No.”

Last Item Attempted Bracket in Timed Tests: The last item attempted by the student is critical in marking the speed and progress of an assessment, and as such is also critical for calculating fluency and “correct items per minute” variables. The bracket itself is a red outline that will be placed around the final grid item attempted. When using a timed test, there are 3 possible scenarios for how the bracket will be placed:

1. It will be placed automatically if the test has been autostopped (if Autostop activated). Once the final incorrect item needed to trigger the autostop has been selected, the screen will flash red to cue the assessor to stop the child, and the bracket will appear over that item.
2. Time expires. The screen will flash red to cue the assessor to stop the child, and the user will be prompted to place the bracket over the last item attempted.
3. The assessor presses the ‘Stop’ button. This should only happen if the child has attempted the final item of the grid (or if the test needs to be restarted due to an error/distraction or for practice). As such, the screen will not flash red as the assessor will know to stop the child. As the assessor should only be stopping time when the final item has been attempted, the red bracket will automatically appear over the final grid item.

To save your changes in the Subtest editor, click “**Done**,” to save and return to the Assessment Builder, where you can then edit the next subtest or add a new page/subtest to your assessment or go back to the Wizard home screen to run and verify your work.

When you use this subtest, you will see a Start button and a grid of empty grey cells. The enumerator will tap “Start” to start the timer counting backwards from 60 seconds (or whatever time you have designated) and to see the items in the grid appear. The default result for each grid item is “correct”. To identify an item as “incorrect”, the enumerator simply touches an item (to self-correct, they just touch the same item again to de-select it). If a child has completed all items before the time is up, the enumerator will hit the “Stop” button to stop the timer and for the system to record the remaining time (12 seconds in the example below).

letters

help

Here is a page full of letters of the alphabet. Please tell me the SOUNDS of as many letters as you can—not the NAMES of the letters, but the SOUNDS. For example, the sound of this letter [point to A] is "AH" as in "APPLE".

Let's practise: tell me the sound of this letter [point to V].

If the child responds correctly say: **Very good, the sound of this letter name is "VVV."**

If the child does not respond correctly, say: **The name of this letter is "VVV."**

Now try another one: tell me the sound of this letter [point to L].

If the child responds correctly say: **Good, the sound of this letter is "LLL."**

If the child does not respond correctly, say: **The name of this letter is "LLL."**

Do you understand what you are to do?

When I say 'Begin,' please sound out the letters as quickly and carefully as you can. Tell me the sound of the letters, starting here and continuing this way: [Point to the first letter on the row after the example and draw your finger across the first line] If you come to a letter you do not know, I will tell it to you. Otherwise I will keep quiet and listen to you. Ready? Begin.

Start

12

T	h	r	Z	f	E	P	S	W	
X	G	C	K	X	S	A	b	v	u
Y	t	J	n	a	Q	F	x	d	y
j	o	G	H	g	p	i			

Stop

12

Restart

Input mode

Mark Last attempted

Next

The enumerator then selects the last item attempted (shown below with red border around the letter "a") and then moves on to the next subtest by tapping "Next":

Letters

help

Here is a page full of letters of the alphabet. Please tell me the SOUNDS of as many letters as you can—not the NAMES of the letters, but the SOUNDS. For example, the sound of this letter [point to A] is "AH" as in "APPLE".
 Let's practise: tell me the sound of this letter [point to V].
 If the child responds correctly say: **Very good, the sound of this letter name is "VVV."**
 If the child does not respond correctly, say: **The name of this letter is "VVV."**
 Now try another one: tell me the sound of this letter [point to L].
 If the child responds correctly say: **Good, the sound of this letter is "LLL."**
 If the child does not respond correctly, say: **The name of this letter is "LLL."**

Do you understand what you are to do?
 When I say "Begin," please sound out the letters as quickly and carefully as you can. Tell me the sound of the letters, starting here and continuing this way. [Point to the first letter on the row after the example and draw your finger across the first line] If you come to a letter you do not know, I will tell it to you. Otherwise I will keep quiet and listen to you. Ready? Begin.

Start										12
T	h	r	Z	f	B	i	P	S	W	
X	G	C	K	X	S	A	b	v	u	
Y	t	J	n	a	Q	F	x	d	y	
j	o	G	H	g	p	i				
Stop										12

Restart

Input mode

Mark: Last attempted

Next

Input Mode: When an enumerator starts a grid test, the “Input Mode” shown at the bottom of the grid is set by default to “Mark” – meaning that the enumerator is marking or scoring the test results. When the test is stopped (either by autostop, the time running out, or by the enumerator tapping “Stop”), the “Input Mode” switches automatically to “Last attempted” as shown by this button’s highlighted state in the screenshot above. After the test, the enumerator can manually switch the “Input Mode” back to “Mark” in the event that he or she needs to make corrections to the test results after the time has been stopped. This should only be used in cases of an erroneous click or difficulty in marking a grid item during the timed test – it is not recommended to have enumerators rely on this mode regularly to score their results. No changes to test marks can be made to items positioned sequentially after the grid item marked “Last attempted” – this is a logic built into Tangerine to avoid confusing results (e.g. prevents cases where item 10 is marked as the “last attempted” and item 13 is marked “incorrect”).

6.6 Survey pages (Untimed and Question and Answer Subtests)

There are two levels of editing survey pages: the “top” level identifies the survey subtest’s name and instructions and the “lower” level involves editing the separate questions that comprise the survey itself. To begin, click on “Add Subtest” at the bottom of the Assessment Builder page and select an option from the ‘Survey’ category of subtest types. In the example below, we have chosen the “Student Information” prototype, which is perhaps the most common and generic form and can be customized to suit many types of survey needs.

Assessment Builder

Name

T Number Identification
 Addition Level 1
 Gr Letter Name Knowledge
 Timed Syllable Identification
 X Location
 School Location
 Dc survey
 Student Information
 Rational Counting
 Missing Number
 St Number Discrimination
 A Word Problems
 Pupil Context Interview
 Su Listening Comprehension
 Initial Sounds
 Untimed Reading Comprehension
 Addition Level 2
 Dictation
 Phoneme Segmentation
 Timed Reading Comprehension

Please select a subtest type

Name

Add Cancel

Once you have selected and named your survey type, clicking “Add” will create this subtest and attach it to your assessment. You should now see your new survey subtest appear in the list of subtests within your assessment. Click on the edit icon to begin adding survey questions.

In addition to settings that are shared with grid subtests (such as “Skippable”, “Enumerator help”, “Student Dialogue” and “Transition Comment”), other settings fields available in the Subtest Editor for surveys are as follows:

Preferred Font: If you need to embed a specialized font not supported by a Unicode library, you can request that the font be loaded into Tangerine (email: contact@tangerinecentral.org); the font name would then be entered here. At present, the loaded fonts are: gentium, andika, padauk, zwekabin, rabiati, and Mondulkiri.

Action on Display: Command line logic for skipping an entire subtest (if entered on Subtest Editor) or question (if entered on Question Editor). See section 6.8 for more details.

Autostop After N Incorrect: In some instances, you might need to configure a survey-type test instrument to stop automatically after a specified number of consecutively incorrect responses. This option will “autostop” the subtest by skipping the remaining questions in the subtest without requiring skip logic commands entered for each question. This feature is used in the ‘Initial Sounds’ subtest of the EGRA, shown below, which consists of 10 survey-type questions, each of which has three response options: Correct / Incorrect / No response. To be considered an “incorrect” response that counts towards the *consecutive* errors needed to trigger this type of autostop, incorrect response options must be assigned a value of 0 or 9.

Focus Mode: If you would prefer that your survey questions appear on the screen one at a time (as opposed to the normal view showing the entire subtest), activate Focus Mode. This will generate two additional buttons allowing users to move between each survey question: Previous Question and Next Question.

Linked to grid: If your survey question subtest is tied to a grid subtest (e.g. reading comprehension), you would then identify specific that linkage by selecting the appropriate subtest in this drop-down menu.

If your survey subtest is linked to a grid which has been autostopped, the linked survey subtest will be skipped over automatically.

By clicking “**Add Question**,” you will be taken through adding questions as per the process outlined below. Under the “Prompt” box, you will indicate the question the enumerator will ask the student (e.g., “What language do you speak at home?”). The “Variable name” is what you will name your question, and which will be included in your database (e.g. s_q1 – for student questionnaire question 1).

Once you have added a question, you can edit it in further detail by clicking the edit button next to the question itself. Note that, similarly to the Assessment Builder, once you have a number of questions added in the Subtest Editor, you can drag and drop them around by selecting, holding, and dragging the three horizontal grey bars to their left. The following screenshot provides the view of the Question Editor, where all remaining information for each question can be provided.

Back

Question Editor

Subtest Student Information
Assessment Copy of Ghana Oral Assessment of Literacy

Done

Variable name

Gender

Prompt

What is the student's gender?

Note to enumerator

Skip if ⓘ

Custom validation

Valid when ?

Error message

Action on display ?

Skippable

Yes No

Items attempted required on linked grid

Question Type

single multiple open

As can be seen above, the 'high level' data on the question has already been prefilled in the appropriate text fields for 'Prompt' and 'Variable name'. You will now add additional information, including hints, answer formats and answer categories for non-open questions.

Note to Enumerator: This is information for the enumerator and will not be read out to the participant as part of the question. This is helpful when answer categories are of the Correct, Incorrect, No Response format and the enumerator is to score, and also when you want to provide format instructions such as

“Please write the answer as a number” or “Please write the date as month-day-year”. When the answer options are set to “correct / incorrect / no response”, we strongly advise supporting the enumerator by providing the correct answer as a hint, e.g., for a reading comprehension question, the question prompt may be “What is Modou’s little brother’s name?”, the note provided may be “Samba”.

Skip Logic, Custom Validation, and Action on Display: These fields are used to define conditions and rules for when your question will be skipped, and what kind of responses you will allow. For taking advantage of Skip Logic and the conditional formatting fields located, see section 6.7 on Validation and Skip Display Logic.

Items attempted required on linked grid: This setting automatically determines the feasibility of asking Reading Comprehension questions based on the number of words attempted in the Oral Passage Reading subtests (i.e. the participant is only answering questions related to portions of the test which he/she has attempted). The timed grid page with short story items is typically followed with the Reading Comprehension Subtest, which is a series of survey questions. To link the two, select the Oral Passage Reading grid subtest from the drop down box “Linked to grid” in the Subtest Editor for the Reading Comprehension Subtest.

To link individual questions to grid locations: For each survey question, you need to specify the cut-off for the number of words attempted in order for each survey question to appear. That is, provide the number of words a child will need to have attempted in order to be presented each comprehension question. For example, imagine we are creating survey questions, which are tied to a previous Reading Comprehension grid whose first sentence is, “Charlie has two brothers and three dogs” and our first survey question is “How many dogs does Charlie have?” We would only want to ask this first question if the last item attempted on the Reading Comprehension grid is at least 7, as “dogs” is the 7th word on that grid subtest. Enter this number in the field for “Items attempted required on linked grid” “ in the Question Editor (see below). As noted above, be sure to include a hint to show your enumerator what the correct response should be:

Items attempted required on linked grid

Question Type

single multiple open

Question Type: Next, you will want to select the answer format and type of question. You have three options for this:

- Single:** You can present as many options as necessary; users will be allowed to select only one response.
- Multiple:** You can present as many options as necessary; users will be allowed to select as many responses as appropriate.
- Open question.** If you select open question, Tangerine will automatically insert a text field underneath the question in the enumerator view for the enumerator to type in the participant’s response. These can be accompanied by hint text in order to help standardize response types and formats, as shown in the screen shot below.

For single and multiple choice questions you will need to define the answer options and assign values to these. To do so, select the single or multiple button, and then select “Add Option” which will generate fields for “Label” and “Value” as shown below. For each answer option, provide the Option Label (e.g., Correct, Incorrect, No Response, English, Kiswahili, French, etc.) depending on question, as well as the Value (e.g. “0” for Incorrect, “1” for Correct and “99” for No Response [without quotation marks]).

Fill from template

Options

≡

Label

Correct

Value ⓘ

1

⊘

Add option

Fill from template: Under the drop-down menu “Fill from template” visible in the above screenshot, we have provided several common examples of question answer (read “option”) types, such as Binary Agreement for “Yes / No” questions, and a “Month” template which currently provides 12 options with month labels and sequential values. These are meant to be illustrative and helpful but each of these can be tailored to suit your specific needs.



When scripting long surveys, break the survey down into sets of 10-15 questions and start a new survey subtest (select e.g. the “student information” subtest from the dropdown – (see 6.6 Survey Pages) for each set of questions. Reason: Tangerine will automatically save results and responses collected each time the Next button was hit to move from one subtest to the next. Should the assessment get interrupted or the application freeze for whatever reason, enumerators will be able to resume (see Completing an Assessment) on the subtest page following the completed set of questions for which results were saved by hitting Next.

6.7 Validation and Skip Logic

Options

The screenshot shows a vertical list of three options for a survey question. Each option is contained in a box with a hamburger menu icon on the left. The first option has a 'Label' field with 'Standard 1' and a 'Value' field with '1'. The second option has a 'Label' field with 'Standard 2' and a 'Value' field with '0'. The third option has a 'Label' field with 'Other' and a 'Value' field with '3'. To the right of each option box is a 'Skippable' button, represented by a circle with a diagonal line. At the bottom of the list is a button labeled 'Add option'.

When designing any subtest or survey question in Tangerine, you are presented with the option of having that component be “Skippable.” This is represented by an option labeled, “Skippable” with “Yes / No” buttons adjacent to this field. As noted, this option is set to “No” by default so as to force an enumerator to validate each subtest or question before proceeding (i.e. to validate by not allowing any unintentional or purposeful skipping of any portion of the assessment).

In some instances, however, you may wish to ask questions that are not relevant for all participants. Consider an example, “*Which class were you in last year?*”, for which answer options could be something like those presented in the screenshot on the right, each assigned its own value:

In order to capture further details for those who answer with “Other”, we would add a follow-on question and use skip logic to define when it should be asked. Thus in order to capture additional information from the example above, “What class were you in last year?” these are the steps we would take:

1. Create a second question designed to capture details on for those who responded with “Other”. In this case, the follow-up question is “*If ‘other’ please specify:*”
2. Enter a command in the **Skip If** field tying this question to the preceding question.

Variable name

Prompt

Hint

Skip if example: ResultOfQuestion("maze1") isnt "2"

Skippable

In Tangerine’s data outputs, questions are identified by their variable name, and answers by their assigned values.

1. As such, in the screenshot above, we have told Tangerine to skip asking this question if the answer to our first question, “What class were you in last year?” was not “other.”
2. Remember to set this question type to “open” so that your enumerators can freely enter text to capture information.
3. Test out your skip logics to ensure that your enumerators are able to capture detailed information and to ensure that they are not being required to answer questions for which there may not be a response (e.g. forcing an open field response for “other” when this was not the answer selected from the first question).

Example Skip Logic commands

The following are examples of texts you could enter into the “Skip If” field for questions of the “Single” or “Multiple” type. For questions of the “Open” type, please see the section below on **Custom Validation**. Note that you can enter more than one piece of logic for each question by using the “or” command to tie your question to more than one answer or more than one previous question. For each answer or question you want to incorporate into your skip logic, you will need to repeat the “ResultOf...” command in order to have Tangerine recognize your instruction.

In the following examples, the *variables* are those inside the parentheses (e.g. “ht_13”), and the response option *values* are those outside of the parentheses (e.g. “4”). Each of these command scripts would be entered into ‘*Skip If*’ field, with the spacing exactly as shown:

For Skip Logic responding to “single” type questions:

ResultOfQuestion(“ht_13”) is “12” or ResultOfQuestion(“ht_13”) is “4”

ResultOfQuestion(“ht_4”) isnt “2”

For Skip Logic responding to “multiple” type questions:

“5” in ResultOfMultiple(“t_13”) or “4” in ResultOfMultiple(“t_13”)

not (“2” in ResultOfMultiple(“m_fav”)) and not (“1” in ResultOfMultiple(“m_fav”))

For Skip Logic responding to questions on *previous* Sub-tests:

ResultOfPrevious(“ht_1”) is “1” or ResultOfPrevious(“ht_2”) is “2”

ResultOfPrevious(“ht_5”) is “2” and ResultOfPrevious(“ht_6”) is “0”

ResultOfPrevious(“ht_1”) isnt “3”

For Skip logic responding to a range of acceptable answers in Open questions:

```
parseInt(ResultOfPrevious("s_8" ) ) < 10 (for connecting to a variable from a previous subtest) par-  
seInt(ResultOfQuestion("s_8" ) ) < 10 (for connecting to a variable from the same subtest)
```

For additional examples, please see Appendix 5 - Sample Skip Logic Commands.

Please Note: If you are having trouble configuring your skip logics but still wish to capture follow-up information, you might consider switching the “Skippable” default from “No” to “Yes.” Making any subtest or question “Skippable” can provide ample opportunity for your enumerators to collect information; however there are risks associated with this as you might not know if your enumerators skipped questions because they were not relevant to the participants or if they simply made a mistake in their data collection.



The Skip Logic commands used in Tangerine are case-sensitive and space-sensitive. Therefore, you must type precisely the name of the variables which you want to reference, and you must pay attention to the capital and lower-case letters of all portions of your commands. Similarly, the quotation marks used in your commands cannot be “hooked” but should be straight up and down, such as in the examples above. You can write skip logic commands using either “is” or “isnt” – where possible, try to use commands that use “isnt” as these questions will only appear once the response option they depend on is selected. Commands using “is” will result in follow-up questions that initially appear on the screen but only disappear when the response option(s) they depend on is not selected.

Custom Validation

For “open” type questions, which provide a text box for enumerators to type in their responses, Tangerine provides you some control over the range of responses you will accept. To include Custom Validation for an open question, find the box below in the Question Editor:

Custom validation
Valid when 

Error message

In the above example, the text entered into the “Valid when” field is written to ensure that enumerators are only allowed to enter numerical values of a certain range (i.e. no written words like, “three” and no numbers outside of the range are allowed). The “Error message” field is the text that will be shown to your enumerator should he or she enter characters that do not satisfy the validation requirements.

You can also specify text strings as allowable answers in open fields. Here are some additional examples for the **Valid when** field:

```
@answer > 0 and @answer < 50 or @answer == “0” or @answer == “NA”
```

```
@answer > 50 and @answer < 100 or @answer == “refused”
```


@answer.length is 3

Note that any text string which is identified as allowable is case-sensitive. Applied to the above examples, responses of “Refused” or “na” would not be accepted by the software, as the custom validation has been set to allow “refused” and “NA”.



“Save” regularly by pressing ‘Done’ to avoid losing your work, and before adding another answer option or going “Back” to the Subtest Editor with the list of questions you have already added for this page of your assessment.

6.8 Subtest Skip Logic and ‘Action on Display’

Tangerine can be configured to deliver more sophisticated survey flows, including automatically skipping entire subtests, automatic survey early stop logic, and dynamic question prompts.

These settings make use of *CoffeeScript* and, to be used beyond the examples below, may require some programming expertise to fully utilize. The examples and in the Sample Skip Logic table found in the Appendix, however, are basic enough to be copy-pasted and edited to fit other surveys, and can tremendously cut down on enumerator error and interview times. In the Tangerine Wizard, there are two places to enter this script; the “**Action on Display**” (AOD) field that can be found on both the Subtest Editor and the Question Editor screens.

Automatically skipping entire subtests

To skip an entire subtest based on the results of a previous question, you will need to know the *variable* name of the “independent question”, and the data *value* of the “independent answer” – both of which will be located in a prior subtest of your assessment. In the example below, the subtest is skipped when the answer option value was 0 in variable B28. This AOD script would go into the AOD field on subtest level (recall that commands are case- and space-sensitive):

```
@skip() if ResultOfPrevious(“B28”) is “0”
```

Assume the subtest we want to skip contains several questions about a student’s classroom experience. Assume B28 refers to the question “Are currently going to school?”, where answer options are “Yes”, with data value “1”, or “No” with a data value of “0”. The above skip logic, entered into the ‘Action on Display’ field of the **Subtest Editor**, would skip the entire subtest with the classroom questions for students who answered “No” (when data value was “0”, in B28).

The inverse operation can also be applied, that is, the subtest is skipped for any answer option chosen – except for the one with data value 0:

```
@skip() if ResultOfPrevious(“B28”) isnt “0”
```

Other examples include:

```
@skip() if ResultOfPrevious(“age”) > 5
```

In this scenario, the variable “age” refers to the age of the child. The data value “5” can either be a simple number entry in an open question format, or a selected age from answer options provided for, say, ages 0-10. This would skip the entire subtest if the child is older than 5.

Dual-condition statements are also possible, e.g.:

```
@skip() if ResultOfPrevious(“B28”) is “0” or ResultOfPrevious(“age”) < 7
```

In this case the subtest could be skipped for two reasons: first, if, using our examples from above, the child is not going to school (B28=0), or if the child is below the age of 7 (age<7).

You can also skip subtests based on performance in previous grid tests. If you would like to skip a subtest when a child has performed poorly on an entry level test, a command such as this might be useful:

```
@skip() if ResultOfGrid("add") < 5
```

In this case, if the child has fewer than 5 correct answers to a previous grid whose variable is “add”, the subtest will be skipped.

Automatic survey early stop logic

At times, it may be necessary to abort or stop an interview before all questions have been asked. This could happen if the participant is not providing consent to participate, and “No, stop”, was chosen in the consent subtest (see section 4.4). As only one “consent” type subtest can be included in each assessment, at times it is necessary to have a second consent cut-off in the middle of a survey subtest. An example could be a household survey designed to ask about children’s schooling, but the participating household has no children going to school.

The applicable script would go into the AOD field in the ***Question Editor*** of the item after the trigger question. For this feature to function, “Focus Mode” must be enabled on the subtest in question:

```
@parent.parent.abort() if ResultOfQuestion("children_in_school") is "0"
```

Please Note: For some types of Action on Display commands to work, Focus mode (see section 6.6) needs to be activated for this subtest. If you are experiencing trouble with your AOD commands, please check the Sample Skip Logic Commands in the Appendix to see if Focus Mode must be enabled.

Dynamic question prompts

If an interview asks many questions from the participant about a third person (e.g. from the head of household about one of her school aged children), it would most likely enhance the interview experience to ask questions in a personalized manner, using the child’s name. First, the name variable (the child’s name) would need to be captured, such as with the question “*What is the girl’s name?*” Let’s assume the variable for this is “name” and the answer option an open text field for the enumerator to type in the name provided. The enumerator would type in, “Ana.”

Subsequent questions can be personalized by customizing the following script into the ***Action on Display*** (AOD) field for every applicable question. Assume one of the next questions is about the age of the girl. Then the prompt for the question in the Tangerine Wizard may read “What is the age of [girl’s name]?”, and the following should be inserted into the AOD field of this same question:

```
@setPrompt "What is the age of #{ResultOfPrevious("name")}?"
```

Remember, if the dynamic question is referring to an earlier question from within the subtest use the command “ResultOfQuestion.” If the command is referring to a question from a separate subtest, use the command “ResultOfPrevious”. When conducting the interview on the tablet, Tangerine will automatically pull the entry that was made for the question with variable “name”, and plug it into the above question prompt for the enumerator to read out: “What is the age of **Ana**?” Note that the AOD command will replace any other text you have entered into the ‘Prompt’ field.

In another example, we may ask about the grade level the girl is in. In the Tangerine Wizard, the prompt for the question may read “What grade level is [girl’s name] enrolled in?”, and the AOD script for this question would read:

```
@setPrompt "What grade level is #{ResultOfPrevious("name")} enrolled in?"
```

Using the above example when conducting the interview, the enumerator would automatically see the question prompt as “What grade level is **Ana** enrolled in?”

6.9 Randomizing Subtests

Subtests



Test Survey	survey			
Date and Time	datetime			
Consent	consent			

Add Subtest

Options

Random Sequences ?

0 - Test Survey

1 - Date and Time

2 - Consent

In certain circumstances you may wish to consider randomizing some or all of the subtests in your assessment. This can be helpful in trying to minimize the order (fatigue) effects of having assessment subtests consistently appear in the same order. Tangerine automatically assigns a number to each of your subtests, based on its order in your Assessment Builder page. As you can see in the example to the right, the numbering starts at zero from the top of the list of subtests and sequences down the list, with numbers appearing below “Random Sequences”:

In this example, “Test Survey” = 0, “Date and Time” = 1, and “Consent” =2. In order to randomize the order of appearance for these three subtests, enter all of the possible sequences for these components into the text box appearing just below the list of subtests. In this example, to randomize subtests 0, 1, and 2, we would enter into the text box all of the possible variations for this sequence, namely:

0, 1, 2

0, 2, 1

1, 0, 2

1, 2, 0

2, 0, 1

2, 1, 0

Tangerine will cycle through the order of subtests as your enumerator performs assessments.



If you are working with long lists of subtests, consider visiting www.random.org to have that website generate random integer sequences for you, tailored to your needs in Tangerine. Verify that your assessment is*

randomizing properly by running through several assessments.*

7. Duplicating and Deleting Assessments

To create a copy of an existing assessment, e.g., when conducting a mid-term assessment using the same instrument as for the Baseline, the quickest way is to use the “Duplicate” function for the specific assessment in the Tangerine Wizard home screen. While at the home page, click the orange arrow next to the assessment you want to duplicate, and then hit “Duplicate”



A second assessment will appear titled “Copy of...” and the name of the assessment duplicated (the ‘Download Key’ of the new copy will be unique). You can now edit the new assessment, change its name, and keep it as a separate assessment (with a separate database) without having to create a new assessment from scratch.

The ‘Duplicate’ function is also useful if you simply want to clean up your results file. Your duplicate copy will not carry prior results, nor will its results file contain variables which had been deleted from your original assessment.

Please Note: If you have already loaded instruments onto your devices and then use the Wizard to create additional questions and subtests, you should always create a duplicate assessment and add your new questions and subtests there. Once you have made your edits, re-load your devices with the updated duplicate version so as to ensure a fresh data file for your results.

Deleting an assessment

To delete an assessment from the Wizard home screen, click on the orange arrow next to the assessment you wish to delete. Then, click the orange circle icon; an option will appear and ask you to confirm deleting the assessment. Press “Delete”. Please note that deleted assessments cannot be recovered.

▼ Test1



Confirm


Delete

Cancel

8. Accessing, Downloading and Interpreting Data

Once data has been uploaded from the tablets to your Tangerine server, you can access the data in the Tangerine Wizard. Log in to your server and navigate to the applicable group, click the carrot “>” found to the left of your



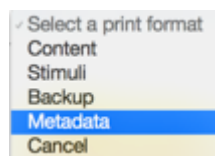
assessment, and hit the “Results” icon, . On the next screen you will find the list of all results that have been uploaded for this assessment with a date and time stamp, and also a button labeled ‘CSV’.

Clicking the ‘CSV’ button will prompt your browser to begin downloading the results file. The CSV file will contain every piece of data that has ever been generated in your instrument. This will include partial assessments (assessments do not need to be fully completed on tablets in order to be uploaded), as well as data from variables which have already been deleted from your instruments. If you have tested your assessments online in the Wizard and opted to save those results to your server, those data will also be present in the CSV file.

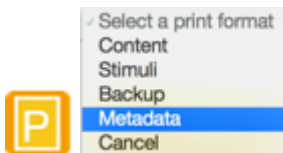
8.1 Interpreting Data

The CSV file is a widely used format for data which can be processed by a range of programs including: Microsoft Excel, OpenOffice, and most statistical analysis packages (STATA, SPSS, R). The results file structure is as follows:

enumerator	Variable 1	Variable 2
Jane_Doe	(value response)	(value response)
John_Doe	(value response)	(value response)



Each line of the data file represents one data entry for your instrument. The values displayed under each variable correspond to the values you have assigned to each response option when designing your instrument. The sequence of the variables corresponds to the order in which data was stored on the server. This order may or may not match the order of variables in your assessment. For example, if you created some variables, collected data on the assessment, but later added other variables or re-ordered some subtests, the order of variables in your CSV file may no longer align with your instrument as it appears on devices. You can obtain a printout of your instrument’s response labels and their corresponding value labels by selecting the ‘Metadata’ printout option:



1. Click the Print button below your instrument.
2. Select the ‘Metadata’ option.

8.2 Default and Automatic Variables

By default, Tangerine will create some variables in your data, regardless of the types of subtests and values you have designed. In addition to these default variables, Tangerine will also automatically generate some variables when certain types of subtests are used (such as Date/Time and Grid tests). These include the following variables:

Variable Name	Description
enumerator	The user name active when data was entered.

Variable Name	Description
start_time, end_time	An EPOCH time stamp automatically generated at the start/end of each,subtest.,See section on ‘Converting,timestamps’ below.
order_map	The sequence of subtests,administered (for instruments using random sequences).
additional_comments	Optional text entered by users at,the end of a given instrument.
[gridvariable]_time_allowed	Will return the maximum time allotted for the given subtest, if a time limit was defined.
[gridvariable]_auto_stop	Will return “TRUE” or “FALSE” to,indicate if autostop triggered for timed tests.
[gridvariable]_time_remain	The number of seconds remaining on,the timer when timed test ended.
[gridvariable]_attempted	The numerical position of the last,grid item attempted.
[gridvariable]_item_at_time	The numerical position of the grid,item captured at a specified time point.
[gridvariable]_time	
intermediate_captured	The number of seconds that have,passed when “item at time” is marked.
year;,month;,date	3 distinct variables generated from,adding the “Date” subtest.

Other variables in your dataset will appear as you named them, and in the order in which they appear in your assessment.

8.3 Converting Epoch/Unix timestamps

In your Tangerine data, you will find a number of ‘timestamp’ variables. There will be one timestamp variable for each subtest in your assessment, and each of these ‘timestamps’ will be taken at the point when the ‘Next’ button has been pressed to advance between subtests, as this is the time when Tangerine is saving your data. The Epoch timestamp value itself is the number of milliseconds that have elapsed between 12:00am on January 1, 1970 (GMT +0) and the point in time when the timestamp was generated.

To convert the Epoch timestamp into a human-readable time, you can either convert it yourself or try an online converter (searching for Epoch or Unix time converter will provide options, remember to indicate that your values are expressed in milliseconds). To convert it yourself using spreadsheet software such as Excel, here is an example of how you would proceed:

A	B	C
timestamp_original	timestamp_conversion	timestamp_end_human
1442035807399	=A2/(60*60*24000) +“1/1/1970”	09/12/2015 05:30:07

In this example, columns B and C contain the same values, but the formatting in column C has been changed to read the value as a date in the format of mm/dd/yyyy hh:mm:ss. Note that the default for Epoch time is that it is on the GMT+0 timezone. You will likely need to adjust the formula in Column B above in order to account for the timezone offset for where your data was collected. For example, if you have collected data in Kampala, Uganda, your data offset is +3 hours, so your Epoch conversion in this example would be: =A2/(60*60*24000)+ “1/1/1970 03:00:00”

What can I do with Tangerine’s timestamps?

1. Check precise duration of an assessment or subtests. If you notice your overall assessment time (per assessor or per group) is lengthy, you may wish to use the timestamps to provide data on which subtests are the most time consuming.
2. Confirm when data was collected. If you have suspicions that your data collector may have manually changed the values on your Date/Time screen against your instructions, you can check for inconsistencies between the Date/Time subtest data and the timestamps, as these cannot be altered by the user unless s/he also alters the

date and time settings on the device.

8.4 Data from Instruments with Non-Latin Characters

Tangerine data is UTF-8 encoded. This is a widely used character encoding for web-based platforms. However, while UTF-8 encoding is common and widespread, you may experience difficulty in viewing your data if your processing application does not properly render UTF-8 encoded characters.

Microsoft Excel's workbooks do not properly render non-Latin characters with UTF-8 encoding. To view and process data from non-Latin character sets (such as Arabic responses written into an open text response question), you must **first** open your Tangerine results file with an application that can process UTF-8 encoded text. If you have already opened your Tangerine CSV file with Microsoft Excel, you will have changed the encoding, and so you will need to re-download your data so that you can view it properly through an application that can process UTF-8. Most statistical packages can process non-Latin characters, so no further action is necessary if you are opening your Tangerine CSV directly into your statistical package.

Apache's OpenOffice spreadsheet application will allow you to properly view non-Latin characters processed with UTF-8 encoding. When opening Tangerine's CSV files with OpenOffice, simply select UTF-8 as the character set.

9. The Tangerine Results Dashboard

The Tangerine Wizard contains a useful tool to manage and monitor data collections. The Results Dashboard is specific to each group. Inside your group, locate the button “**Results**” at the top right to access the Results Dashboard.

Once you are in the dashboard, you are presented with a range of options of how you would like to view basic overview details about your data. Under the “Assessment” option you can choose specific assessments/instruments that you would like to view or you can select “All” and see a basic summary of all of the assessments in your group (see below). “Value used for grouping” lets you decide how you would like to sort the results (see below). Options include sorting by enumerator, assessment name, start time, subtest, school, county, etc. Under “Advanced Options” you have the option to shift the time value by a certain number to handle the correct time zone.

group-tang_test_may13

Assessment: All
Value used for grouping: enumerator

Advanced Options

enumerator	20th May	21st May	22nd May	23rd May
nicky	10	1	2	5

The Results Dashboard will display summary tallies about your data (whether on an individual assessment/instrument or for the entire group, according to how you have sorted it). These numeric values represent the number of fully completed assessments/instruments. A result is considered fully complete when the user has clicked or tapped ‘Save Result’ on the final page of your assessment. Therefore, though data from partial assessments will be available in your results file, assessments which have been only partially completed will not be included in the totals represented in the Results Dashboard.

B. Tangerine on Mobile Devices – the “app”

1. Hardware Selection and setup of Tangerine software

Hardware selection - Minimum requirements

Tangerine requires the following minimum features for use as a data collection tool:

2. Capacitive touch screen (models with resistive screens can also be use, but are not recommended)
3. Android OS [v. 4.0 or higher]
4. HTML 5-capable browser (standards, such as Chrome or Opera Mini preferred)
5. Wi-Fi (b/g/n)
6. 4GB HDD (8GB preferred)
7. 512MB Memory
8. 7 hours battery life (recommended)
9. Ability to install third-party Android applications

Additionally, the following are **desired features**:

- 3G connectivity (if tablet supports a SIM card)
- GPS
- mini-/USB port
- mini-/HD slot
- 7 inch screen
- Ideally below 1lb. in weight

Suggested compatible hardware

A range of tablets/eReaders meet the minimum requirements, including the Archos 70, Barnes & Noble Nook Color and Tablet, Huawei Ideos 7, Samsung Galaxy Tab series, and Asus ZenPads. Several RTI projects have used Nexus, Asus and Samsung models. As of July, 2017, RTI recommends Samsung and Asus models of 7” or larger screens for EGRA/EGMA data collections.

Hardware purchase and delivery

At this point in time many of the recommended devices may not be available for purchase in all countries. If you decide on purchasing a device which in turn needs to be shipped to your data collection site, you should plan significant time in advance to have the purchase delivered and then shipped to the country for data collection (otherwise you may need to hand-carry equipment). For some tablets, you may be required to create a new Amazon.com or gmail account and assign the devices to that account as part of the initial device setup. Two examples of this process are detailed below.

Setting up your Kindle

1. Create an Amazon.com account with a project-specific name and email > address, and a password that is easy to remember; do not provide a > credit card or other payment information for this account
2. When the Kindles arrive, you will turn them on and it will lead you > through a setup process. You will see a message “Welcome [name of > person who purchased Kindles]”. Next to this message there will > be a link that says “Not [name]? Deregister Kindle”.
3. Click on “Deregister”, and re-register your Kindle to the new name > that you have set up.

Note that each Kindle also has a unique email address. In your project inventory, you will want to note the serial number of the device, the Kindle name, and the email address. This information can be found on each device in:



Settings > **More** > **My Account**

You can also manage all of your Kindle devices through the Amazon website: www.amazon.com/myk

Setting up your Nexus

1. Create a gmail account easy to remember.
2. When the Nexus arrive, you will turn them on and it will lead you through a setup process.
3. Register the devices to the gmail address.
4. Turn off location services and do not allow for using location for search results, nor use of GPS: Settings/Location services – uncheck all.
5. Do not set up GoogleWallet when (if) asked when you register the devices.
6. Enable screen rotation which is handy for Tangerine: From inside any application (does not work on home screen), touch the center of the top bar (same level as where wireless icon and time is). You will see a menu open (see screenshot), hit the button with the rectangle in between two brackets next to the dates. It should look like in the screenshot.
7. Make sure Bluetooth is set to off, WiFi to on inside Settings.
8. Disable English language spell checker: Settings/Language & input. Untick Spell checker and untick Google voice typing.
9. Before you install Tangerine: Go to Settings/Security, and tick the box next to “Unknown sources”.

Procurement, Importation and Customs

Requirements will vary from country to country and depending on the project. Whether shipping or hand-carrying equipment, there may be import duties and customs clearance to arrange, including export licenses or other documentation. Requirements also differ for tablets, routers and other equipment. It is important to find out what the requirements will be in each situation and consider that for budget and timeline purposes.

Recommended Accessories

You should plan to purchase a **protective case** for the device – those that feature clip-ins and a zipper to entirely protect the device from dust are best. The protective case also facilitates handling the tablet when administering an assessment or survey. It is also recommended to purchase a **stylus** and an appropriate **soft cloth** for wiping the screen. The only other accessory delivered with the tablet is the power adaptor, which may be a wall plug type different than that of your field site, so you may also need to **outlet adaptors** for your country. The AC/DC charger has a voltage adaptor so there is no need to purchase one.

Mobile Internet Access for Uploading Data

All tablets have an internal WiFi adapter, but may not have a port for inserting a SIM card. However, you can purchase a **mobile Internet hotspot** such as the TP Link MR3040 or a similar device available locally. Mobile network operators (AirTel, Vodacom, Safaricom) often sell mobile hotspot units that can be used to share a single network connection across multiple devices. These devices will allow you to connect a USB dongle with a SIM card and data plan (either for the month of data collection or over a certain amount of data, e.g. 20MB) from a local mobile service provider, to connect to the Internet via 3G (note that in many cases, these devices may not be able to provide Internet access when only 2G networks are available). Then the tablets will connect to the hotspot and be able to access the Internet for synchronizing the Tangerine dataset to the remote server. Many hotspot devices have rechargeable batteries that, fully charged, should yield about 6 hours of constant use and 20 hours of stand-by time.

In the field the devices are usually only switched on when back-up is being conducted for a few minutes every evening. Thus, one full battery charge should last several days for this use scenario.

Tablet Battery Capacity and Charging

Make sure you test the battery power of your device, that may be anywhere between 7 (KindleFire, used) and 9 hours (Nexus 7). Depending on local context, it may be necessary to have options for charging the tablets at the end of every day. Options include: external battery pack, USB-car adaptor, or solar power adaptor. We have tested the USB-car adaptor sold as a tablet accessory, and it took about 4.5 hours of continuous driving to fully charge the battery. With a typical wall outlet, expect that it will take at least two hours to fully charge the tablet. More information on charging devices can be found in Annex 4.

2. Installing Tangerine on your tablet

Installation of Tangerine

Before installing anything, be sure the Android device allows you to install applications that are not provided by way of the Google Play Store. In the settings menu of the device, look for the option “Allow installation of (Third Party / Unknown Source) Applications” and set this option to “Yes” or “On”.

To install Tangerine on the tablets, ensure you are connected to the web.

- From the main group page on the Tangerine wizard (see section 2 of this manual) click on the button labeled ‘APK.’ APK stands for Android Package, it is the file type associated with apps to be installed on Android devices. The Tangerine server will take a few minutes to process your request, during which time the Tangerine icon in the top left will spin and a green indicator bar will flash. Once your APK is ready for download a window will appear, presenting you with a URL (web address) which is the link to download an APK containing the Tangerine app and all of the active assessments in your group.
- If you will be installing Tangerine and all of your instruments on many devices, you may wish to consider shortening the URL for your APK, by using a shortening service such as *bit.ly* or similar. [Alternatively, you can install the APK file from Tangerine onto your laptop computer and transfer it onto tablets over the USB cable. To do so, you may need to install another Android application that allows you to transfer files in this way. We recommend ES File Explorer, which is available for free through the Google Play store.]
- Verify that the device is connected to the Internet, open a browser on your tablet device and enter the URL (or shortened URL) into the web address line. You will know that the file is downloading when you see a small number (1) next to the device name in the upper left-hand corner, or find there a small arrow pointing down, or find the app listed in the download section of your device. You may also immediately notice the words “Starting download...” flash along the bottom of the browser. Touch the number/arrow/listing and install the app. You can also find your downloaded APK in the downloads folder on your device.
- When it is done installing, find your downloaded APK file and click “Open” and follow any additional steps to confirm installation. You will know when it is finished you will see a Tangerine login screen. You have now installed and launched the application. From now on, you can get into the application by clicking on the Tangerine icon on the home page or within the Apps menu.



*Your Tangerine APK contains settings unique to your group at tangerinecentral.org. As such, after installing your APK, you will be able to use assessments and upload results related only to your group. Recall that even if you have archived an instrument online, an instrument from a tablet can still upload data to an archived instrument. **PRIOR TO***

DATA COLLECTION: VERIFY THAT ALL ASSESSMENTS HAVE LOADED CORRECTLY AND COMPLETELY ON EVERY DEVICE (HAVE USERS CHECK ALL ITEMS AND SUBTESTS).

3. Logging into Tangerine on mobile devices

When Tangerine opens, the user will see this login / signup screen:



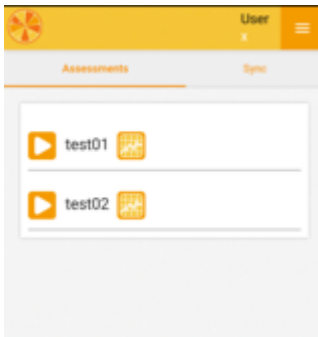
Notice there are two tabs in the login screen: **Login** and **Sign Up**. Login is for users who have already registered a user account on the device and on the .apk file that is currently installed. **User accounts which you have created on other tablets, or on the online wizard, or on other versions of the .apk file, are not recognized as they have not been created within that tablet application's memory.**

Sign Up. New users should tap the 'Sign Up' tab to establish an account on the tablet. It is best to create user names and passwords that are simple to remember. Keep a record of your username and password in a safe place (such on your mobile phone or in your wallet). It is best to choose a username that can be associated with an individual. Typically, RTI will require data collectors' usernames to consist of the first letter of their first name, followed by their surname. So user John Smith's username would be: **jsmith** and his password would be something simple and easy for him to recall.

Login. Users who have already signed up on the tablet (and on the version of the app that is installed) can enter their credentials to login. If a user has lost or forgotten the username or password, they can simply create a new login with the 'Sign Up' tab. All data from all users on a tablet are uploaded – there is no data lost when a user needs to create a new account.

4. Instruments on Tablets

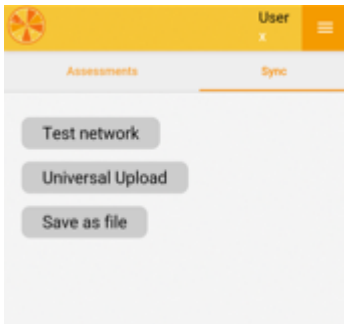
After you have installed the Tangerine .apk file and logged into the tablet, you will see all of your active assessments from your online Wizard group listed on the home screen. The Tangerine app home screen *for enumerators* may look like this:



Under “Assessments” tab, the enumerator is able to see the different assessments that can be run. Clicking the “Run”



button will start the assessment. The “Results”  button will bring the enumerator to a screen that shows the data collected thus far, and allows the option for interrupted assessments or surveys to be resumed.



The “Sync” tab gives the enumerator the option to send their results to the Tangerine server where they were created. The “Universal Upload” button in the middle of the screen allows the enumerator to upload all of the new data from all of the assessments on the APK. **This will upload all new data from all users who have entered data on that device, not just from the user who is currently logged in.**

In case you are unable to perform a universal upload, due to poor connectivity, you can also use the “Save as file” option which will create a back file with the results data that can be imported on the server. To make use of this function, tap the “Save as file” button, and Tangerine will indicate the file directory on the tablet where the file has been created. Connect the tablet to a computer, open the tablet’s internal storage and navigate to the indicated backup file location to extract the file. The backup file can now be emailed to your Tangerine server administrator for merging into your results files.

5. Training and Data Collection

Using Tangerine during training, pilot and data collection

You may use the steps outlined earlier in this manual to prepare the assessment and the tablets for enumerator training. The enumerators can practice entering and saving data, and when you are ready to begin the actual data collection and no changes are needed to your assessments, you can just filter your database by date and know which tests were part of training and which were part of the actual data collection. Similarly, you may use Tangerine for piloting early versions of your instruments.

Alternatively, to create separate versions of each assessment (i.e., training version, pilot version, final), duplicate your test in the Wizard by clicking on the duplicate icon. Then give the new assessment a different name. Results will not

be carried over under duplication, but make sure you check the new instrument to confirm that questions and subtests have been fully duplicated.

Tangerine Training Topics

Based on RTI's experiences with Tangerine enumerator training for EGRA/EGMA work, no additional time should be needed with a Tangerine data collection compared to a paper-based data collection. The training agenda should be the same, with only an additional one to two hours set aside for introducing the tablet hardware and familiarizing enumerators with basic navigation and operation of the hardware and software. Key topics need particular emphasis:

1. Properly logging in and out (including fully powering off the tablet) on a daily basis.
2. Selecting the correct school in the location subtest and what to do if the school is not available in the pre-loaded list (e.g. because of a last minute school replacement)
3. Hitting the start button of timed subtests only when the child attempts the first item (not when the enumerator says "Begin").
4. Connecting to a wireless network and synchronizing/uploading results.

An example training agenda and topics that are important to cover is included in Annex 1.

Inter-rater reliability (IRR)

Procedurally, the process of inter-rater reliability does not change with Tangerine. Please refer to the best practices cited in the EGRA toolkit available at www.eddataglobal.org or follow those of your project/organization. To summarize the minimum recommended steps:

- Make sure an Internet connection is available and that all enumerators have a tablet (so make sure you have extra tablets to cover the extra assessors).
- Prepare an assessment for doing IRR (make sure it does NOT contain randomization).
- As you begin the IRR exercise, ask assessors to confirm that their tablet: shows their username in Tangerine, and shows the correct date and time.
- Have everyone listen to the same stimuli and record their responses, while someone on the team records the 'gold standard' reference model against which they will be compared.
- When the stimuli are finished, ask assessors to enter "IRR" as text in the 'Additional Comments' field found at the end of every assessment.
- Upload saved results from this assessment(s), including the 'gold standard'.
- Download the.csv file from the Tangerine Wizard for analysis, and conduct the analysis as you would normally.

Preparing for final data collection

As noted, you may want to create separate versions of each assessment for key stages in the data collection process: enumerator training, instrument piloting and final data collection. **This will be very helpful in managing your data files and instrument changes.**

We have found the most practical way to make an initial, full version of the instrument and append the words "training / pilot / final" in the assessment names. For training purposes, you may wish to create single-subtest assessments, such as "Letter Sounds" which allow users to repeatedly focus on just one subtest during training. It is handy to have these for practice, to avoid enumerators having to "click through" too many unnecessary subtests.

During the training we usually find one or two things to edit in the instruments before piloting. We do this by duplicating the full training instrument, editing the necessary items and renaming it to append "pilot" or "final" to the assessment name.

6. Entering data on tablets

Launch your assessment. Before entering data, you must open the assessment that you wish to conduct. Some countries may have more than one assessment developed, i.e., EGRA (potential several in various languages), EGMA, and/or SSME instruments.



Click the run icon to begin collecting data on any given instrument.

Types of input. Input in Tangerine will either be done by tapping response items or entering text. Below are descriptions of the different types of input; shading in red indicates where you should apply pressure to the screen (either with stylus or finger) to activate the element.

- **Buttons.** Most input is done through buttons, including advancing from screen to screen (“Next” button), assigning a student ID, starting or stopping the timer, or indicating incorrect items in an assessment grid. To use a button, simply touch the button in the center and release with a short tap or a firm, but short, press. On grid item pages, the button will turn green to indicate a wrong answer. You can click a second time to unmark the item and it will return to grey. You cannot ‘unclick’ the start and stop buttons. Once you press “Start” or “Stop”, clicking it a second time will have no effect if the timer has already started. After clicking “Stop”, you can click a grid item to mark it as the last item attempted. It will be shown in red with a bracket. Clicking it a second time will have no effect, but you can change your selection of the last letter attempted by clicking a different letter.
- **List selection.** The school name field is an example of an automated list selection input type. Users must select items from drop-down menus, the contents of which are filtered based on how this subtest has been designed.
- **Automatically generated fields.** Certain field values are automatically filled in. These are the date and time fields, which are generated according to the date and time that the device is set to when the assessment begins; and the unique student ID field, which is filled in when the assessor presses the “Generate” button. While these fields can be changed manually, assessors should be instructed not to touch them unless necessary (such as when users are entering data collected earlier on paper) - just advance by using the “Next” button.



Timed subtests. On a timed grid page, you will not see the grid items until you press “Start”. When you press

“Start”, the timer will start counting down and you will see the items appear in the grid (see screenshot shown at right). Once you have started you should not stop and restart except for exceptional circumstances. If a restart is needed, there is a button at the bottom of the page for that purpose.

By default, all grid items are “correct”. Mark items “incorrect” by tapping on that item, causing a line to be struck through that grid item. Re-mark them correct by tapping a second time. Use the asterisk button on the right-hand side of the grid to mark a whole row incorrect. Re-tap the line wrong icon to correct the line all at once. There are three ways to end a timed subtest:

- *the program will autostop*, if that feature has been turned on in the wizard. If this occurs, the screen will flash red, and a message will appear that says “Autostop activated. Discontinue test.” The next step for the assessor is to press “Next” to move to the next exercise.
- *the timer will run out*. If the timer runs out while the child is still reading, the screen will flash red, and a message to “select the last item read” will appear (see below). The enumerator should ask the child to stop reading, then touch the last item attempted by the child when the timer went off. If it is necessary to mark the last item read as incorrect, use the “mode” feature (see below). Otherwise, the next step for the assessor is to press the “next” button.



- *the assessor will click the stop button*. If a child completes all of the grid items before the allotted time runs out, the assessor will stop the timer by using the “stop” button. This will automatically place the ‘Last Item Read’ bracket around the final grid item, as the timer should only be stopped when the child has attempted the final grid item. If need be, the ‘Last Item Read’ bracket could be moved away from the final grid item.





The “Input mode” allows the user to toggle between marking items correct/incorrect and marking the last item attempted. When the timer has started, the buttons are in the “Mark” mode that expects a click to mean “mark this item incorrect”. After the timer is stopped, the mode automatically changes to “Last attempted” and the application expects that the next click will mean “mark this as the last item read” (above). In some cases, the last item read is both the last item AND an incorrect item. Therefore, the assessor must toggle between modes in order to mark the button accordingly. There is also a mode for “Capture item at specified number of seconds.” This is for a special administration case where the subtest may be untimed, but you want to mark the item read at a certain point (i.e., 60 seconds).

Completing an assessment. Upon reaching the end of the assessment (after the last subtest or interview question), you will see a confirmation page that the test has been completed, and an overview of results for each subtest (select the details button on “Subtests completed”). This is for the enumerator’s benefit as a way to verify that all of the subtests were completed.

Users now have the option of adding “Additional Comments” as deemed appropriate. Typically, RTI will ask that users only add comments that are important for analyses (e.g. “I noted this child was a female but it is actually a boy”). Other subjective comments (e.g. “This child did well”) are not encouraged as these will not be taken into account for analysis.

Once users have decided to make a comment or ignore this field, they should press ‘Save Result’ to finalize their entry. This button will then become “Perform another assessment.” You can either press this button to begin assessing

another child on the same instrument, or you can use the Tangerine icon button  in the upper-left hand corner to get back to the list of available tests and start a new assessment.

To check the number of assessments collected, return to the list of assessments. Click on the data/results icon  located to the right of any assessment and you will see a list of assessments conducted by that assessor for that test. Click on each assessment to see details.



Never delete or modify an assessment or APK file from a tablet until you have uploaded all of the data! Remember that copying an assessment online does not create a copy of your results data, only the instrument content.

7. The Student/Participant Identifier

A recommended Tangerine subtest is the “Student ID” subtest, which generates an anonymous and unique student code for each assessment. This facilitates identifying records in your results file. This subtest is designed such that enumerators can click the “Generate” button to produce a student code, but enumerators are not able to manually enter codes for this subtest unless they are entering codes previously generated by Tangerine. Tangerine uses an algorithmic formula to generate IDs, and as such an attempt by an enumerator to enter an ID not conforming to this formula will not be accepted.

For data collections where one student will undergo more than one assessment (e.g. English EGRA, Kiswahili EGRA, and English EGMA), we have suggested preparing index cards that list the various assessments and provide a space to note the student identifier for each child. If a child reaches an enumerator and the index card does not yet provide a student ID number (a 6-digit combination of letters already generated by Tangerine, e.g., XCVRRR), the enumerator must generate a new ID and copy it to the student's index card. Thus, the enumerators of the second/third assessment can look at the index card and just copy the student ID by typing it into the relevant field in Tangerine instead of creating a new one for this student.

Tangerine will then tell the enumerator if a student identifier is invalid. The enumerator shall then double-check the card to make sure she/he didn't make a mistake in the transcription and press “Next” at the bottom of the screen. If Tangerine still doesn't accept the Student ID from the notecard, the enumerator shall click “Generate” to create a new one, and write this new identifier down on the card as well and put a circle around it to indicate the issue. The enumerator of the last assessment for this child should retain the index card to allow for reconciliation of the observations at data cleaning stage.

Tangerine ID's can also be useful for verification of scoring for writing samples. For example, if your assessment includes a Dictation task, you may have trained your assessors to score student writing. You may want to verify the accuracy of their scoring, in which case you can have assessors copy the Tangerine ID for the assessment onto the student's writing sample. You can then collect the writing sample and cross-check the associated Tangerine data.

8. Data Storage, Synchronization and Backup

Data collected with Tangerine will be synchronized/uploaded to the Tangerine server housing the APK file used on the tablet. The server used may be one hosted by RTI or another provider of the open software who has installed it on their server. As Tangerine is open source software, any user has the license-free option of hosting Tangerine on their own webserver if they should wish to. The details provided in this manual refer only to the provision of Tangerine as offered by RTI International. Check local laws and regulations about export of personal data to ensure that your use of Tangerine is compliant.

RTI International does not advocate the collection of personally identifiable information (PII) on Tangerine. The storage of PII on Tangerine's servers may jeopardize individuals if such data were to be provided to authorities with punitive powers or criminal intentions. As such, Tangerine users should take the utmost care in ensuring that data stored on Tangerine does not contain PII and/or that any individuals with the permission to access your data are aware of and compliant with your organization's ethics policy. Please see the Terms of Use on Tangerine's website for further details.

Making backups during data collection

Tangerine works in a browser, but does not require an Internet connection. There are two principal ways to back up your data during fieldwork, but prior to proceeding with the backup options described below, users should first attempt to use an existing Internet connection that will send the data to the above-mentioned Tangerine server. If you have been able to sync your data successfully on the Tangerine server, there is no need to proceed with the backup options below.

‘Sync tablets’

On the home screen of Tangerine, you will find a button labeled ‘Sync tablets’ just below the list of your loaded instruments. Once this button is tapped, Tangerine will begin looking for other tablets using Tangerine which are connected to the same WiFi network. If two or more tablets are connected to the same wireless network, this process will create duplicate copies of all data on all tablets, such that data loss will be mitigated if anything should happen to any of the tablets. This process relies on ‘peer-to-peer syncing’ which is a data transfer protocol not allowed by all wireless networks, depending on their defined security settings. You will see messages appear on the tablet screens indicating the progress and success of this backup. Note that this process will not work between two tablets connected to the same 3G or mobile network – the network connection between the tablets must be provided by a wireless router establishing a local area network.

Full tablet backup / Exporting Tangerine data

A more involved backup than the ‘Sync tablets’ option is to backup all of the data stored within your Tangerine application. This process can be done either by backing up the entire tablet’s data, or by copying only the Tangerine data file. The memory taken up by backing up the entire tablet’s backup will be greater, but the process will be easier to follow than extracting only the Tangerine data file. For more novice tablet users, backing up the entire tablet may be easier to follow. For more advanced users (such as those who may be selected as data collection team supervisors), the steps for exporting only the Tangerine data file may be feasible.

1. Full tablet backup. Prior to data collection, search for and install a Backup application from Google Play store. A search on Google Play will provide several options for this kind of app. As steps within each app vary, you will need to use the application yourself to identify the specific steps necessary for backing up either the entire tablet or, if possible, a selected range of applications. Typically these backup apps will then ask you to identify where you will store your tablet’s backup file – whether on a laptop or an external storage such as an SD card or USB port, if your tablet is equipped with these.
2. Exporting Tangerine data. On your Android tablet, Tangerine stores its data in a file named, “tangerine.couch” which can usually be found in the following directory: Android/appdata/Tangerine. To find the file in this directory, you may need to install a File Manager application, such as the ES File Explorer app (free from Google Play). Once you have a File Manager app installed, open the File Manager app and then find the folder named, “Android”. Thereafter open “App Data” and then “Tangerine” in order to find the “tangerine.couch” file. Once you have located this file, you will need to either store it on a laptop or save it to an external storage option available for your tablet (such as an SD card or USB port).

In the event you should lose or damage tablets and need to merge data from a backup into your larger online database, please contact the RTI Tangerine team for steps on how to send the Tangerine data file from your backup in for processing.

9. Hardware and software maintenance and troubleshooting

Battery maintenance. A new tablet’s battery will typically last for 8 - 10 straight hours of use, or approximately 20-25 EGRA assessments. To fully recharge the battery, you may need approximately 3 hours or more.

Battery life may be lengthened if the tablet’s WiFi and GPS function are disabled and the tablet is powered down when not in use. Reducing the tablet’s screen brightness will also extend battery life. Note the difference between putting in ‘sleep’ mode (screen going black) and powering off by holding the power button of the device and choosing and option like *Shut down* or *Power Off*.

Should the device stop functioning because of low battery, you will not lose the assessments that have already been saved on your device (even if you have not yet synced your data with the online Tangerine server). If this happens while you are with a student or interviewee, however, you may lose some of the data you are currently collecting.

Annex 1: Training Topics and Example Agenda

Training on the use of Tangerine for data collection should be integrated into the regular EGRA training programs, just as use of paper for marking responses would be. Based on RTT's experiences, only an additional hour at most is recommended so that enumerators can become familiar with the hardware and its main functions.

Topic	Details
Tablet basics Suggest 1 hour,orientation, including free,time to practice/explore	- On/Off/Sleep. Tablet care. - Navigation: touch screen vs. stylus; keypad, including how to hide the keypad. - Browser: refresh, enter URL - Settings: check battery life, turn on/off Wi-Fi and verify Internet,connection;,brightness or screen lock,settings; set time and date. - Touch bottom of screen to show menu items (careful not to touch during,testing) - Icons: settings, wireless, battery, on/off
Sign Up, Login/logout Practice subtests Suggest 2 hours, subtest,by subtest, with practice,in pairs.	- Open Tangerine from application list - Register/Sign Up User Name and,Password. - List of tests, opening test options; define each icon. - Start test using the play icon. - Logout Depends on specific assessment(s). General issues to cover: -Button states (disabled,,activated, etc.) - Using the school information autofill - Generating student ID - Start/stop timer - Ending a test and starting a new,assessment
Troubleshooting	- Subtest doesn't respond - Can't find current EGRA version - Errors caused by touching wrong, timer start/stop
Saving/syncing data	Depends on local context

Annex 2: Tangerine Tipsheet

To start:

- Turn on tablet using the exterior buttons on the device
- Locate the “Tangerine” icon (orange square) in your bookshelf or an “Apps” menu
- Touch it once
- Login using your username (or Sign Up if you are a new user)
- Select the test you want from the list of tests. Open it using the orange arrow icon.

To get back to the Tangerine home page (instrument list):

- Click on the round fruit icon in the top left of the screen

To turn on wireless:

- Touch the icons in the top right corner of the screen to active the Wi-Fi, or go via “Settings”
- Touch “Wi-Fi”
- Find the available wireless hotspot or network in the list and tap its name
- Touch “connect”
- Wait for wireless signal icon (without an x) to show up in the list of icons top right

To send test data to the server (EVERY DAY BEFORE LEAVING THE SCHOOL):

- Go to the test list (using the round fruit icon in the top left of the screen)
- Tap “Universal Upload” on Tangerine’s home screen
- Wait for confirmation message, “Results Successfully Synced to Cloud”
- Remember to fully ***Power Off*** all devices at the end of each day.

! Important reminders !

- For timed grid exercises: Press START when the child starts reading. If the child uses the entire minute, the timer will stop automatically and the screen will flash. The next thing to do is mark the last item read. If the child reads everything before the minute is over, press STOP as soon as the last word is read or attempted.
- For timed grid exercised: Touch an item to mark it wrong. Touch it again to mark it correct if the child autocorrects or if you have touched it by mistake. A blue item with a line through it means INCORRECT.
- Do not ever touch any tablet specific navigation bar, e.g. on the Kindle or Nexus at the bottom (with home, arrow, etc.) during a test. During a test you only need the NEXT button to navigate. **YOU CAN NOT GO BACKWARDS OR START OVER** (unless your particular instruments are using the “Back” button).
- Never use the “Input mode” buttons UNLESS you necessary or if must mark the last item wrong. In this case, after the timer stops and you mark the last letter, change ‘mode’ to “mark item” and touch the item to mark it wrong.

Annex 3: Guide to Supervisor Responsibilities during Data Collection

Use the following guidelines to modify or create your supervisor guide.

I. Prior to the start of data collection

- (some of this may have been done already by the principal researcher so you should coordinate closely)
- Create device logsheet, with serial number (and, in the case of Kindles, Kindle name) and any other accessories that go with the device (charger, stylus, cover)
- Develop user agreement and check out form
- Carefully proof-read/pre-test the electronic version of the instrument on one of the devices. Ensure that all subtests, instructions, student dialogue and items are included and correct. Verify the functionality, including: next buttons, last item marking, comprehension questions area aligned with distance read in the text.
- Download your unique Tangerine app (APK file) onto the tablets.
- Verify that each instrument has been fully loaded on the device and all subtests are present.
- Ensure that each device is set to the date and time of the area(s) in which you will collect data.
- Check out each device to an assessor by receiving a signed copy of the checkout form and updating the inventory list

II. Before and during the school visit

- Ensure that each enumerator has the device assigned to them and that it is fully charged.
- Ensure that any alternate devices are charged and ready if needed.
- Before leaving the school, verify the number of assessments saved on each device.
- If a network connection is present, upload the data. Supervisors should understand how to use the mobile wireless hotspots if using tablets that are not enabled with SIM card ports.

III. After each school visit

- If network connection was not present at the school, get to a network connection and upload the data.
- It is up to each project/supervisor to determine whether assessors at the end of the day will be responsible for their device or if the team supervisor will be responsible for all of them. The particular arrangement should be reflected in the device check-out form that each assessor signs.
- Fully Power Off and charge each of the devices.

IV. At the end of data collection

- Collect the devices and return the user agreement, countersigning it to confirm that you have received it in good condition and with any accessories (case, stylus, charger, hotspot router).
- Store the devices in a safe place until returning them to the principal researcher or logistician.

Annex 4: Tips for keeping Tablets Charged

Tablets should be chosen so that they can be used for a full day without running out of power. In order to conserve power the **Wi-Fi should be turned off and the brightness set to the lowest usable level.**

Many places in the world that do not have grid electricity will still have a way to charge mobile phones. The solution is going to be different for each context. In general, tablet users will need to take advantage of power whenever it is available. This could mean plugging it in at a school during use, or at a restaurant during lunch or at a hotel while sleeping.

Even in places where there is no grid electricity, there will probably be a business that uses car batteries to charge phones. We can use this infrastructure to charge tablets for using Tangerine, as cell phone chargers are basically the same as tablet chargers. If you ask someone where to charge a tablet they might be confused, but if you ask them where to charge a mobile phone, they will probably be able to help. The operator of the charging station should be able to provide a place to plug the charger into.



Based on Tom's Hardware, charging times are as follows:

- **9 hour to charge Kindle over USB**
- **3 hour charge via AC power**

Charging times for the Nexus 7 are approximately the same as for the Kindle Fire.

The voltage that a Kindle Fire charges with is 5V, but the current (amperage) varies based on the charging device:

- Typical USB charge supply (from computer) is 2.5W (.5A)
- Normal USB AC Adaptor is 5 W (1A)
- Kindle Fire AC Adaptor is 9 W (1.8A)
- Nexus 7 AC Adaptor is 5 W (2A)



Hence, the fastest charge comes only when a Kindle Fire AC Adaptor (or Nexus 7 AC Adaptor for the Nexus) is used. But this requires AC voltage. We can get this from a hotel or from a car battery with an inverter.

A typical small car battery inverter will put out about 150 watts continuously. This means that a car with an inverter could charge at max 14 Kindles (or Nexuses) without the inverter dying, but it would drain a typical car battery in about 8 hours, unless the motor is running, when it can do it continuously for the cost of fuel.



Voltaic systems has a nice little tablet solution (Spark Tablet Case: <http://www.voltaicsystems.com/spark.shtml>). The \$300 device would need to be in the sun all day in order to charge a single Kindle. If we were never going to have access to power, then this might be a way forward, but it would be expensive. Their V39 USB battery is \$100 and looks pretty good (battery only, no solar panels). They claim 1.8 Kindle charges.

A better solution might be to just have double the tablets for all locations where power is an issue. Every day while the enumerators are enumerating, a designated person (driver perhaps?) is responsible for charging the second set of tablets. This could be done either with an inverter in the car or going to a place with mains electricity or finding a local cell phone charging service.

Annex 5. Additional Examples of Skip Logic Commands

The example strings of text can be copied and pasted into your instruments; you will just need to adjust the variable names and values used so that they conform to your needs. Recall that all spacing and characters are case-sensitive in Tangerine skip commands. The variables shown below in green are examples only and would need to be changed to align with however you have named your own variables. Note: for Question-Level items, you will need to change “ResultOfQuestion” to “ResultOfPrevious” if the initial variable being referenced is contained in a previous subtest than the one where you are entering the skip logic. The same rule applies to response data coming from open questions (i.e. when using `parseInt(ResultOfQuestion)` or `parseInt(ResultOfPrevious)`).

Where to Use	Description	Example	Focus Mode Required?
Subtest Level Action on Display	Skip an entire subtest depending on responses entered in a prior subtest. The prior subtest can be a survey, or location type.	@skip() if ResultOfPrevious("xxx")) is "0"	No
Subtest Level Action on Display	Skip an entire subtest depending on the number of correct responses from a previous grid test.	@skip() if ResultOfGrid("add") < 5	No
Subtest Level Action on Display	Skip an entire subtest depending on geographical factors (e.g. when only some questions are intended for specific states/provinces). The initial variable in this example, "province", would match whichever 'Geographic Level' you have defined in your school location subtest.	@skip() if ResultOfPrevious("province")) isnt "west"	No
Question Level Custom Validation	Require a numeric response within a defined range, and allow one designated value for cases where the data is unavailable or unknown, "888"	@answer > 0 and @answer < 50 or @answer == "0" or @answer == "888"	No
Question Level Custom Validation	Require a numeric response which is allowable when compared with a previous data point (e.g. when total number of books in good condition should not be less than the total number of books available).	parseInt(@answer) <= parseInt(ResultOfQuestion("total_books"))	No
Question Level Custom Validation	Require that a numeric response be the sum or the difference from two prior responses.	parseInt(@answer) == parseInt(ResultOfQuestion("num1")) + parseInt(ResultOfQuestion("num2"))	No
Question Level, "Skip if..."	Skip a question if the answer to a prior question is above or below a specified value.	parseInt(ResultOfPrevious("xxx")) < 10, or, parseInt(ResultOfPrevious("xxx")) < 10	No
Question Level, Action on Display	Display the result of a mathematical operation applied to previous data points, as a question-level hint. Note that the hint / result display must be in a different subtest from the original data points. This particular example is designed to show the teacher attendance rate, as a percentage calculated on days absent vs. total school days.	@setHint "This teacher is present #{parseInt(parseFloat((ResultOfPrevious("school_days")) - (ResultOfPrevious("teacher_absent")) / parseFloat(ResultOfPrevious("school_days")) * 100))}% of the total school time. Please note this with teacher and head teacher."	Yes
Question Level Custom Validation	Require that a response to an open question contain text only (no numeric responses).	not @answer.match(/[^\a-zA-Z]/)	No

Where to Use	Description	Example	Focus Mode Required?
Question Level, “Skip if...”	Skip a question depending on one or more answers to previous multiple-response type questions.	not (“2” in ResultOfMultiple(“grade”)) and not (“1” in ResultOfMultiple(“section”))	No
Question Level Action on Display Question Level, Action on Display	End / Abort a survey or assessment based on the answer value to a previous variable.,E.g. if a parent responds that a child is not in school, automatically end the interview.	@parent.parent.abort() if ResultOfQuestion(“children_in_school”) is “0”	Yes
Question Level, Action on Display	End / Abort a survey or assessment based on the answer value to a previous variable.,E.g. if a parent responds that a child is not in school, automatically end the interview.	@parent.parent.abort() if ResultOfQuestion(“children_in_school”) is “0”	Yes
Subtest Level (Student ID test) Action on Display Question Level, “Skip if...”, and Custom Validation	Hide the ‘Generate’ button of the ‘Student ID’ subtest.,This may be desired if, for example, you are conducting a longitudinal study and want to require that users are entering an ID that has been previously generated by Tangerine. Ask a user to re-enter a Student ID Code that matches their previous entry.,This may be applicable if your users are entering ID’s that were previously generated (such as with a longitudinal study).	\$(“#generate”).hide() In the ‘Skip if...’ field, enter this: console.log(“#{ResultOfPrevious(‘participant_id’)}”); false; In the ‘Custom Validation’ field, enter: @answer == ResultOfPrevious(“participant_id”)	No No