



# Global Adult Tobacco Survey (GATS)



## Programmer's Guide to General Survey System



# **Global Adult Tobacco Survey (GATS) Programmer's Guide to General Survey System**

September 2020

## **Global Adult Tobacco Survey (GATS) Comprehensive Standard Protocol**

---

### **GATS Questionnaire**

Core Questionnaire with Optional Questions  
Question by Question Specifications

### **GATS Sample Design**

Sample Design Manual  
Sample Weights Manual

### **GATS Fieldwork Implementation**

Field Interviewer Manual  
Field Supervisor Manual  
Mapping and Listing Manual

### **GATS Data Management**

Programmer's Guide to General Survey System  
Core Questionnaire Programming Specifications  
Data Management Implementation Plan  
Data Management Training Guide

### **GATS Quality Assurance: Guidelines and Documentation**

#### **GATS Analysis and Reporting Package**

Fact Sheet Templates  
Country Report: Tabulation Plan and Guidelines  
Indicator Definitions

#### **GATS Data Release and Dissemination**

Data Release Policy  
Data Dissemination: Guidance for the Initial Release of the Data

## **Suggested Citation**

Global Adult Tobacco Survey Collaborative Group. *Global Adult Tobacco Survey (GATS): Programmer's Guide to General Survey System*. Atlanta, GA: Centers for Disease Control and Prevention, 2020.



## Acknowledgements

### ***GATS Collaborating Organizations***

- United States Centers for Disease Control and Prevention (CDC)
- CDC Foundation
- Johns Hopkins Bloomberg School of Public Health (JHSPH)
- RTI International
- World Health Organization (WHO)

### ***Financial Support***

Financial support was provided by the *Bloomberg Initiative to Reduce Tobacco Use* through the CDC Foundation with a grant from Bloomberg Philanthropies.

---

Disclaimer: The views expressed in this manual are not necessarily those of the GATS collaborating organizations.



# Contents

---

Chapter	Page
<b>1. Introduction .....</b>	<b>1-1</b>
1.1 Overview of the Global Adult Tobacco Survey .....	1-1
1.2 Use of this Manual .....	1-2
1.3 Hardware and Software System Requirements.....	1-3
<b>2. Case Management and Transmission System.....</b>	<b>2-1</b>
2.1 Case Management System.....	2-2
<b>3. GSS Engine .....</b>	<b>3-1</b>
3.1 Overall Design.....	3-1
3.2 Example 1—Programming a Small Questionnaire .....	3-4
3.3 Example 2—Using GSS to Program the GATS Household Questionnaire .....	3-16
<b>4. GSS Tool Kit .....</b>	<b>4-1</b>
4.1 Questionnaire Designer (Quex Designer).....	4-2
4.2 Edit DB Tables .....	4-8
4.3 Case File Tools .....	4-9
4.4 Data Aggregation .....	4-10
4.5 Help .....	4-22
4.6 Program Options .....	4-22
4.7 Exit .....	4-23
<b>5. Files and Structures.....</b>	<b>5-1</b>
5.1 CMS Database.....	5-1
5.2 Questionnaire Database Tables.....	5-7
<b>6. Configuring the GSS for a Handheld Device.....</b>	<b>6-1</b>
6.1 How to Configure an Android Device.....	6-1
6.2 GSS Files .....	6-1
6.3 How to Configure the Case Management System.....	6-2
6.4 How to Configure the GSS.....	6-7
6.5 Language Customization .....	6-7
<b>Appendix A: GSS Programming Language.....</b>	<b>A-1</b>
<b>Appendix B: Review of the Main Data Tables .....</b>	<b>B-1</b>
<b>Appendix C: Frequently Asked Questions .....</b>	<b>C-1</b>
<b>Appendix D: Acronyms List.....</b>	<b>D-1</b>

## Exhibits

---

Number		Page
Exhibit 1-1.	Sample Configuration of GSS System.....	1-3
Exhibit 3-1.	GSS Engine Screen Elements.....	3-3
Exhibit 3-2.	GSS Engine Screen Types .....	3-4
Exhibit 3-3.	Individual Questionnaire Subset .....	3-5
Exhibit 3-4.	Screen Information: Questions Table .....	3-8
Exhibit 3-5.	Question Text: Texts Table.....	3-9
Exhibit 3-6.	Answer Text and Codes: Answers Table.....	3-10
Exhibit 3-7.	Initialization Data: INI Table for Survey Form .....	3-11
Exhibit 3-8.	GSS Engine Special Instructions Options .....	3-12
Exhibit 3-9.	GSS Screen Properties.....	3-13
Exhibit 3-10.	Household Questionnaire .....	3-17
Exhibit 3-11.	Screen Information: Questions Table .....	3-25
Exhibit 3-12.	GSS Implementation of the GATS Household Questionnaire: Texts Table .....	3-28
Exhibit 3-13.	GSS Implementation of the GATS HQ: Answers Table.....	3-31
Exhibit 5-1.	Purpose of Key Internal Tables of CMSDB.db3 .....	5-2
Exhibit 5-2.	CMS Database Table: AddressLOG.....	5-3
Exhibit 5-3.	CMS Database Table: DU .....	5-4
Exhibit 5-4.	CMS Database Table: DUEvt .....	5-5
Exhibit 5-5.	CMS Database Table: EvtDef.....	5-5
Exhibit 5-6.	CMS Database Table: INI .....	5-5
Exhibit 5-7.	CMS Database Table: Messages .....	5-6
Exhibit 5-8.	CMS Database Table: Notes .....	5-6
Exhibit 5-9.	CMS Database Table: Responses.....	5-6
Exhibit 5-10.	CMS Database Table: TrainingCases .....	5-7
Exhibit 5-11.	Purpose of Questionnaire Database Tables.....	5-8
Exhibit 5-12.	Questionnaire Table: Answers.....	5-9
Exhibit 5-13.	Questionnaire Table: INI .....	5-9
Exhibit 5-14.	Questionnaire Table: Questions .....	5-9
Exhibit 5-15.	Questionnaire Table: Texts.....	5-10
Exhibit 5-16.	Questionnaire Table: QxQ .....	5-10

Exhibit 6-1.	Android Folder Structure for GSS within /.../gats .....	6-1
Exhibit 6-2.	CMS Initialization Table, the INI Table in CMSDB.db3.....	6-2
Exhibit 6-3.	Data Fields of the Cases Table (CMSDB DU).....	6-5
Exhibit 6-4.	List of Fields/Variables for CaseFile Table .....	6-6
Exhibit 6-5.	GSS Initialization Table, the INI Table .....	6-7
Exhibit 6-6.	Messages Table .....	6-8
Exhibit A-1.	Examples of Statements .....	A-3
Exhibit A-2.	Functions Available in QLogic Sentences .....	A-3
Exhibit B-1.	DU Table—Table of Case-Level Records for Each Unique Data Collection Activity .....	B-2
Exhibit B-2.	Responses Table—Table to Store Respondent Answers .....	B-3
Exhibit B-3.	DUEvent Table—Table to Record Status Codes for Data Collection Activities or Events .....	B-4
Exhibit B-4.	Sample Event Data .....	B-5
Exhibit B-5.	Sample Responses Data .....	B-6



# 1. Introduction

---

Tobacco use is a major preventable cause of premature death and disease worldwide, with approximately 1.4 billion people age 15 years or older using tobacco<sup>1</sup>. Furthermore, more than 8 million people die each year due to tobacco-related illnesses<sup>2</sup>. If current trends continue, tobacco use may kill a billion people by the end of this century, and it is estimated that more than three quarters of these deaths will be in low- and middle-income countries<sup>3</sup>. An efficient and systematic surveillance mechanism is essential to monitor and manage the epidemic.

The **Global Adult Tobacco Survey (GATS)**, a component of Global Tobacco Surveillance System (GTSS), is a global standard for systematically monitoring adult tobacco use and tracking key tobacco control indicators. GATS is a nationally representative household survey of adults 15 years of age or older using a standard core questionnaire, sample design, and data collection and management procedures that were reviewed and approved by international experts. GATS is intended to enhance the capacity of countries to design, implement and evaluate tobacco control interventions.

In order to maximize the efficiency of the data collected from GATS, a series of manuals has been created. These manuals are designed to provide countries with standard requirements as well as several recommendations on the design and implementation of the survey in every step of the GATS process. They are also designed to offer guidance on how a particular country might adjust features of the GATS protocol in order to maximize the utility of the data within the country. In order to maintain consistency and comparability across countries, following the standard protocol is strongly encouraged.

---

**GATS manuals provide systematic guidance on the design and implementation of the survey.**

---

## 1.1 Overview of the Global Adult Tobacco Survey

GATS is designed to produce national and sub-national estimates among adults across countries. The target population includes all non-institutionalized men and women 15 years of age or older who consider the country to be their usual place of residence. All members of the target population will be sampled from the household that is their usual place of residence.

GATS uses a geographically clustered multistage sampling methodology to identify the specific households that Field Interviewers will contact. First, a country is divided into Primary Sampling Units, segments within these Primary Sampling Units, and households within the segments. Then, a random sample of households is selected to participate in GATS.

---

**The GATS interview is composed of two parts: *Household Questionnaire* and *Individual Questionnaire*. These questionnaires are administered using an electronic data collection device.**

---

---

<sup>1</sup> World Health Organization. WHO report on the global tobacco epidemic, 2019: Offer help to quit tobacco use. Geneva, Switzerland: World Health Organization; 2019. <https://apps.who.int/iris/bitstream/handle/10665/326043/9789241516204-eng.pdf?ua=1>

<sup>2</sup> GBD 2017 Risk Factor Collaborators. Global, regional, and national comparative risk assessment of 84 behavioural, environmental and occupational, and metabolic risks or clusters of risks for 195 countries and territories, 1990-2017: a systematic analysis for the Global Burden of Disease Study 2017. Seattle, WA: Institute for Health Metrics and Evaluation; 2018.

<sup>3</sup> Mathers, C.D., and Loncar D. Projections of Global Mortality and Burden of Disease from 2002 to 2030. *PLoS Medicine*, 2006, 3(11):e442.

The GATS interview consists of two parts: the *Household Questionnaire* and the *Individual Questionnaire*. The *Household Questionnaire* (household screening) and the *Individual Questionnaire* (individual interview) will be conducted using an electronic data collection device.

At each address in the sample, Field Interviewers will administer the *Household Questionnaire* to one adult who resides in the household. The purposes of the *Household Questionnaire* are to determine if the selected household meets GATS eligibility requirements and to make a list, or roster, of all eligible members of the household. Once a roster of eligible residents of the household is completed, one individual will be randomly selected to complete the *Individual Questionnaire*. The *Individual Questionnaire* asks questions about background characteristics; tobacco smoking; electronic cigarettes; smokeless tobacco; cessation; secondhand smoke; economics; media; and knowledge, attitudes, and perceptions about tobacco.

## 1.2 Use of this Manual

The purpose of this document is to provide an overview of the General Survey System (GSS) and to help users understand the software tools that make up the GSS. This document includes an overview of the technical architecture, the business architecture, and the requirements for the system. Detailed architecture and design models, or references to them, may also be included where appropriate. The document is intended for information technology (IT) staff who will support the GSS and for individuals who will program the questionnaire within the GSS.

The GSS Tool Kit was developed to facilitate the administration, collection, and management of survey data. It is designed to run on a Windows-based computer (PC), and files produced for data capture run on Android-based handheld devices. Currently the GSS is available for three different mobile platforms—MS Windows Mobile-based handheld computers, Windows-based laptop, and Android-based handheld computers (smartphones or tablets). The software system is designed to support field data collection activities where Field Interviewers (FIs) collect data using touchscreen-enabled handheld devices. This manual describes the function and use of the Android version of GSS.

The GSS software consists of three main programs, each dedicated to a specific function:

1. **CMS:** A Case Management System (CMS) that allows users to manage the caseload and transmit data.
2. **GSS Engine:** A specific module that displays questions on the device and collects data. The GSS Engine system was designed to allow most, if not all, of the specifications for a data collection task to be done by simply describing the attributes of the data to be collected and specifying the flow of data collection steps in a single metadata repository (a SQLite database and its data tables).
3. **GSS Tool Kit:** A developer's menu system that organizes the access to the PC-based components of GSS.

*Exhibit 1-1* provides a diagram of the system programs and how they interact. Two of the major programs—CMS and GSS Engine—run on the handheld device that the FI uses to capture data. The GSS Tool Kit runs on a desktop or laptop PC. The exhibit shows the FI link to the central systems as wireless Ethernet, but other network connections to the Internet are available, including cellular

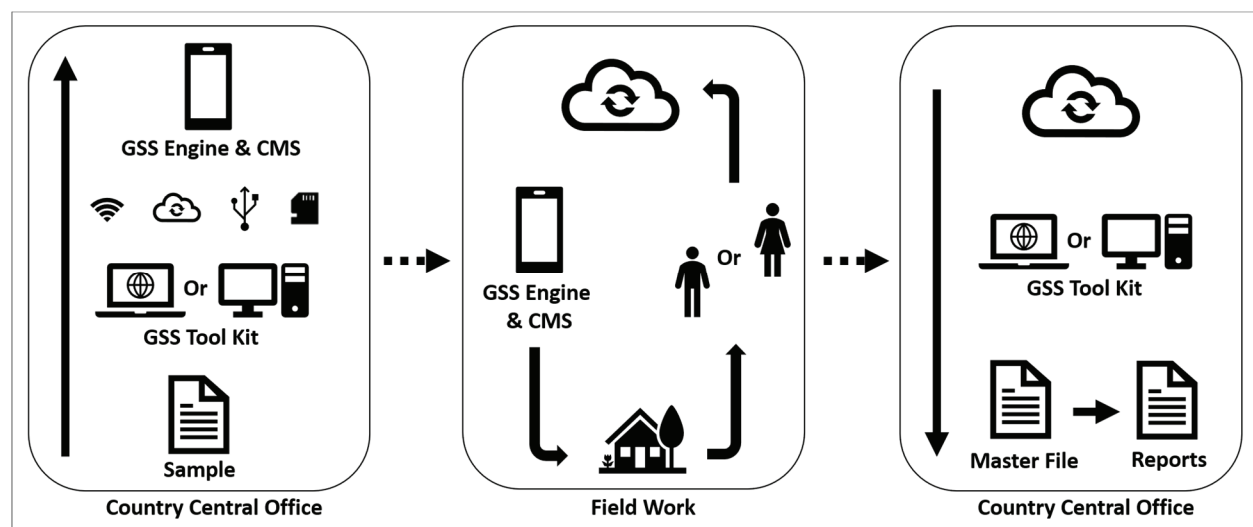


communication if available on the handheld device and wired devices through universal serial bus (USB) ports. The specific communication links available depend on the features of the selected handheld device, but almost all handheld devices have USB ports that will connect to Ethernet devices, almost all have wireless Ethernet, and if using smartphones, cellular-based links will be available. The system also supports file transfer protocol (FTP) transmissions over the network channels if an FTP server is available.

The network communication link allows the FI to do the following:

- Receive new cases.
- Upload data collected during the day via nightly or other periodic transmissions, including:
  - Questionnaire answers,
  - Status codes of work in progress,
  - Case notes, and
  - Corrections or updates to locator addresses.
- Receive updates to software and database tables (questionnaires only).

**Exhibit 1-1. Sample Configuration of GSS System**



### 1.3 Hardware and Software System Requirements

The hardware platforms and software requirements for the programs are as follows:

#### Android device for Survey Data Capture—GSS Engine and CMS

- Hardware
  - Android 4.0 or later handheld device
- Software
  - Android 4.0 or later

- SQLite database

## Desktop PC— GSS Tool Kit

- Hardware
  - Standard Windows-based PC
- Software
  - Windows 7 or later
  - MS Office with MS Access
  - MS.Net Framework
  - SQLite database

### 1.3.1 Other Software and Hardware Requirements

In addition to the software listed above, users may need the following desktop software:

- **Secure FTP client.** This optional software can be used for transmitting data files between data collection sites to a central location.
- **Spreadsheet application (e.g., Excel).** Users may need to create spreadsheets to summarize recruitment outcomes and track any data issues.
- **SQLite data base management tools to view, edit, and create SQLite databases.** Several free or open source tools are available for both the PC and the Android.

### 1.3.2 GSS Software

Each site will receive copies of the files for the following:

- Android data collection applications (CMS, GSS Engine), a *Household Questionnaire* (HQ), and an *Individual Questionnaire* (IQ) in the native country language(s) and in English for collecting interview data specific to the country's questionnaire.
- GSS Tool Kit containing Windows OS PC-based programs for creating/ adding/modifying local questions in the GSS, managing GSS databases, building required support files, and viewing GSS data files. Because changes to the questionnaire could alter the system specifications (e.g., variable names, response values and formats, consistency checks), the questionnaire cannot be modified without the approval of the GATS Questionnaire Review Committee (QRC).

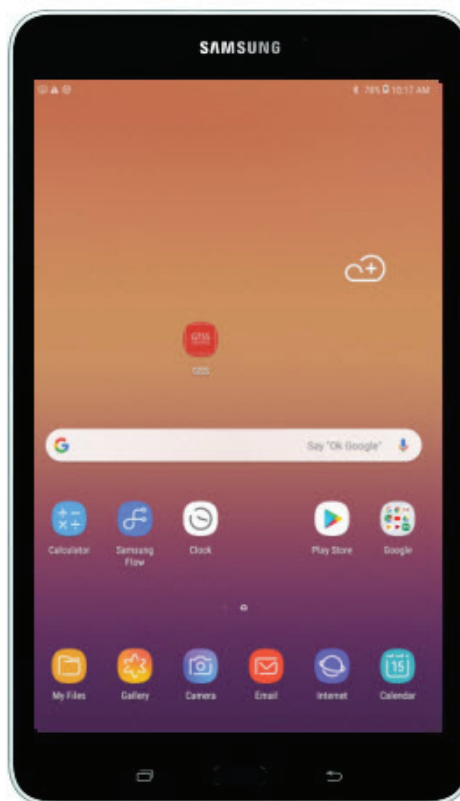
## 2. Case Management and Transmission System

---

GSS users (typically Field Interviewers) see the CMS as the first software module they use. It is the software interface to the GSS Engine and, hence, to data capture. The CMS is used to manage the workload of cases on the data collection device. The CMS is a password-protected system. Users must enter a login password to access the system each time the CMS is started. During logon, passwords are not displayed; they are hidden by masking characters. The system keeps track of failed login attempts and notifies the user with messages at specified intervals.

All of the screenshots and final testing of this version of the GSS Android-based programs have been done on a Samsung Galaxy Tab A (Model SMT-280). This device runs version 5.1.1. The screenshot below displays the home screen of the Android device with the CMS icon installed in the main menu.

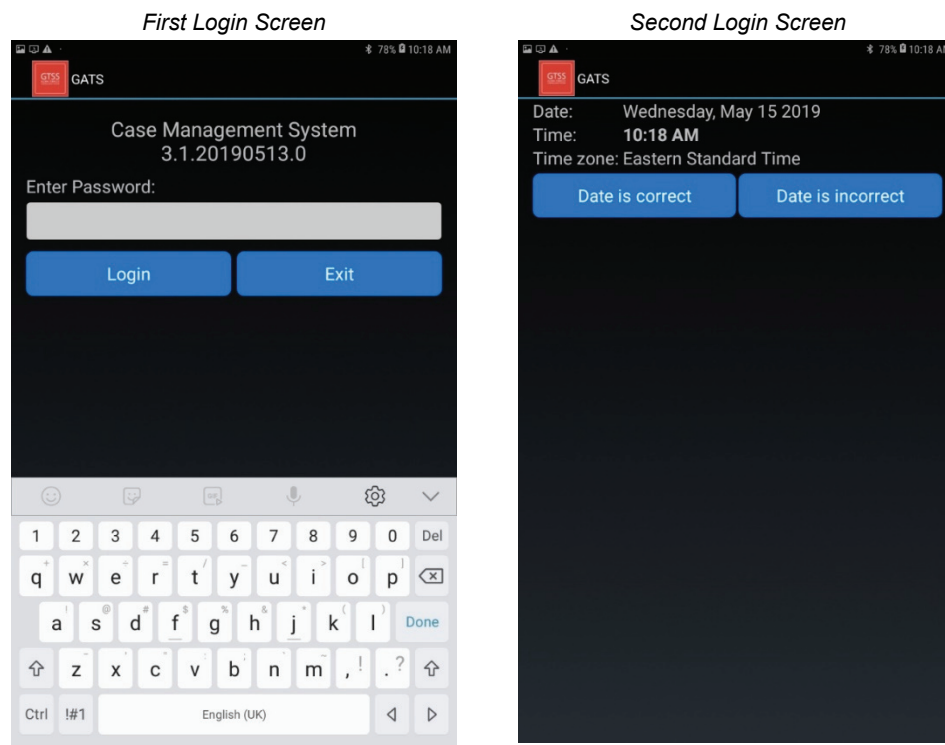
*Samsung Galaxy Tab*



The user simply taps the GATS icon (the orange GTSS icon) to start the first of the GTSS programs, the CMS. This action brings up the CMS login screen. The FI must enter a password to activate the CMS system. As a security feature the CMS implements a login failure action. After five failed login attempts, the CMS is locked and cannot be started without an unlock code, which must be supplied by supervisory staff. The code is available from a GSS Tool Kit utility. The FI must contact the Field Supervisor to obtain this unlock code, which is valid only for the day it is issued.

The CMS login screens are shown below. The first login screen displays the current version number of the CMS. Enter your password and click "Login." The next login screen displays the current date, time,

and time zone. The user should confirm these before logging in. If they are incorrect, the user should reset these using the "Date is Incorrect" button to access the Date and Time settings for the device. Reset the time and/or date and click "Done."



The CMS functions are described briefly in the next section.

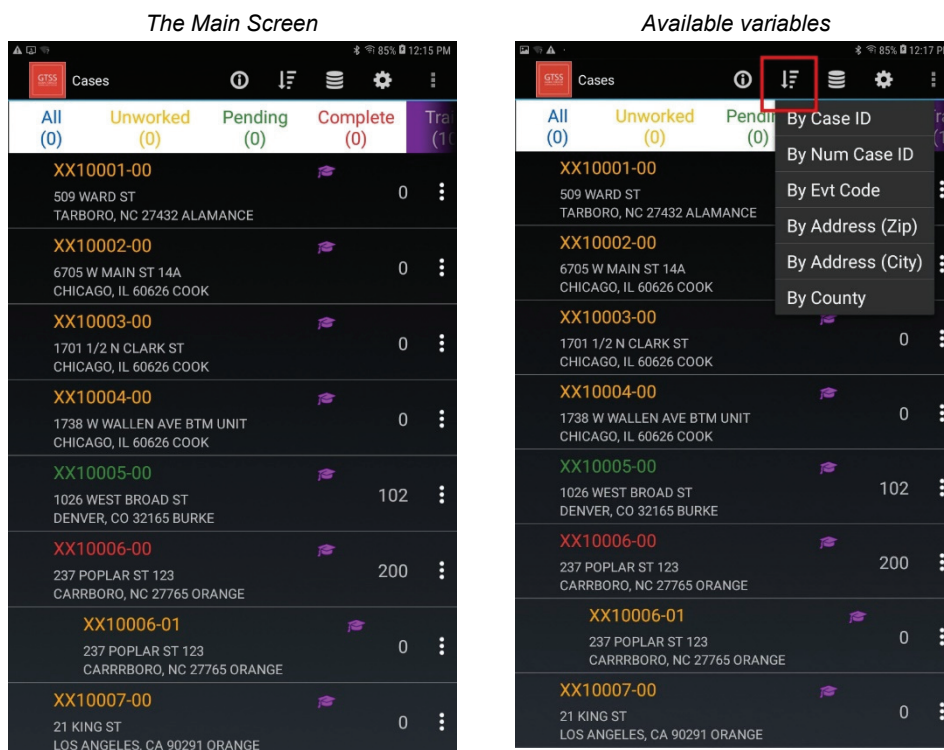
## 2.1 Case Management System

The purpose of the CMS is to allow FIs to manage, view, and transmit their case assignments. The following sections describe the major menus of the CMS that facilitate those goals.

### 2.1.1 Working on a Single Case

When the CMS begins, the program shows all active cases available to the FI on the data collection device. The list of cases should contain all the necessary information for an FI to locate each household and to conduct field work. From this list of households, FIs can plan strategies to execute their assignments.

The CMS main screen, the Select Case screen, is shown below on the left. In this sample screenshot the CMS has been loaded with a set of 10 training cases; only the first 8 are visible. To help FIs organize the households in their assignments, this list may be filtered to display only certain types of cases (All, Unworked, Pending, Complete and Training), and may also be sorted using the variables in the case grid, such as Case ID, Event Code, City, ZIP Code, and County (as seen in the screenshot on the right). Note: The mortarboard cap located next to the Case ID indicates that this is a training case.



Each data block in this screen describes one case. The nine characters in the top left corner of each block is the Case ID. The one- to three-digit number in the top right corner of each block is the case status code. The lines of information under the Case ID are address or locator information for the case. The Case ID may be one of three colors to reflect its current status:

- Amber—indicates the case is pending but has not been worked.
- Green—indicates case is pending and has been worked; at least one event has been entered.
- Red—indicates case is final, either a completed interview or some other final status.

To view a case, touch the item and hold momentarily. A drop-down menu will appear with the following selections:

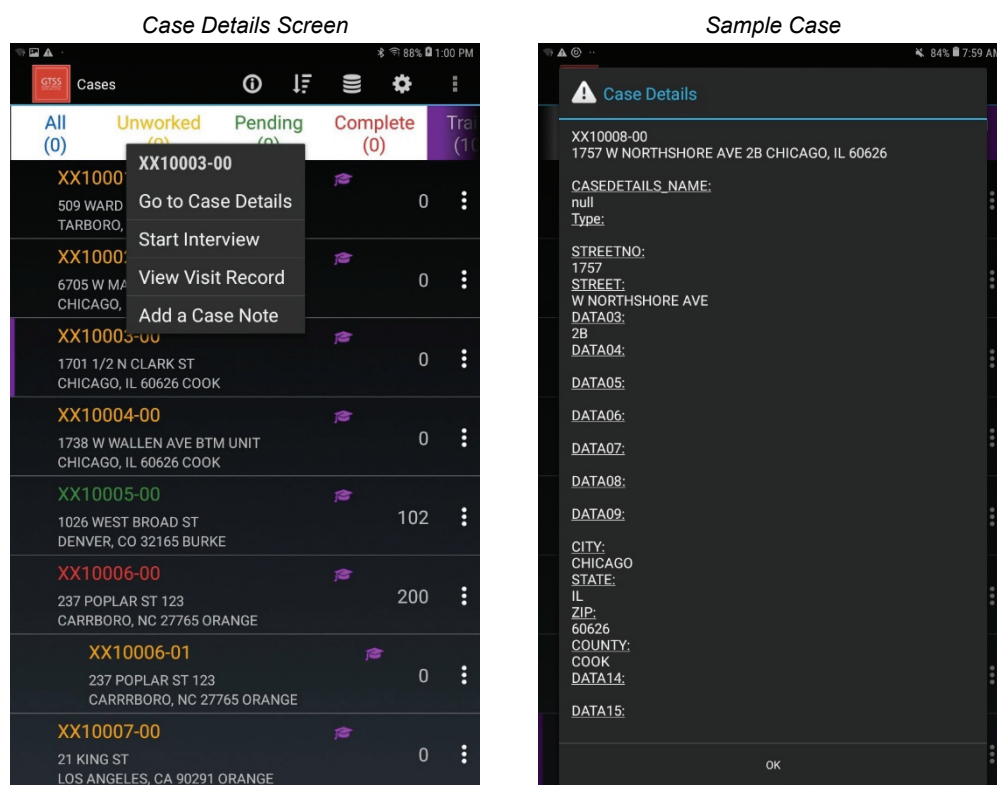
- "Go to Case Details" allows you to view full details on the case.
- "Start Interview" initiates the interview process with the participant.
- "View Visit Record" displays the visits that have been made for this case and allows you to enter a new record.
- "Edit Address" allows you to edit the address.
- "View Address Changes" displays a history of previous changes to an address.
- "Add a Case Note" allows the user to add additional notes related to the case.

Each action is described in a separate section below.

## Case Details

The Case Details screen allows the FI to view data that has been collected. The screen contains all items that are in the Case File for a case. This screen can be modified using options in the INI table. (See *Chapter 6* for more information on the INI table in the CMS database.)

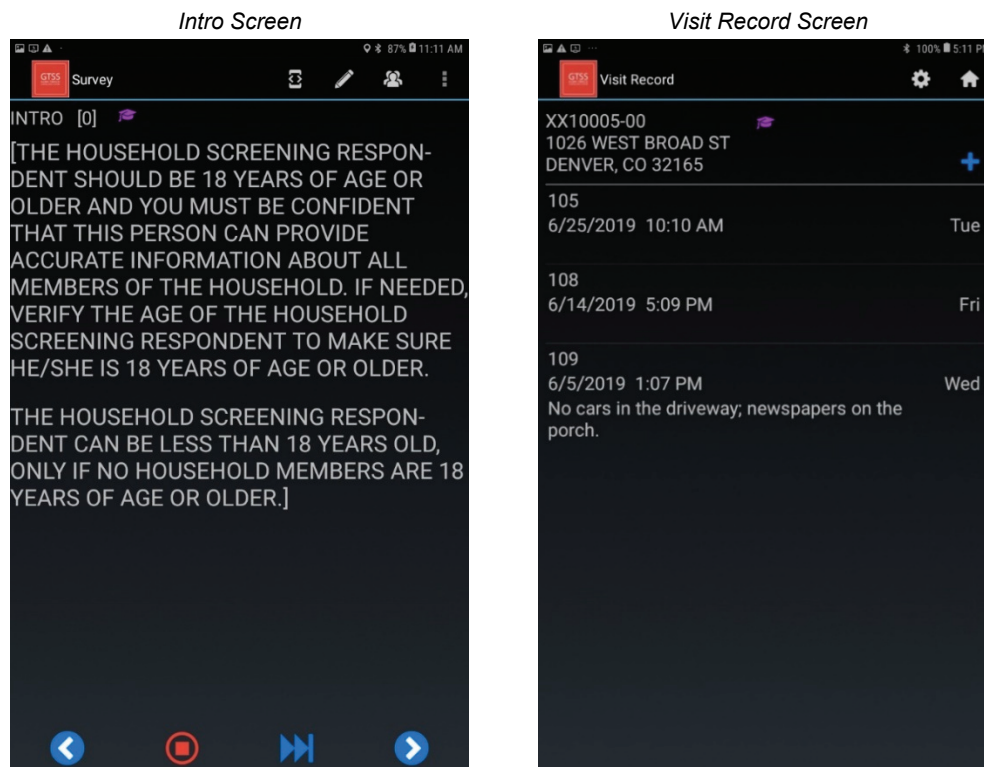
Below is a Case Details screenshot for a training case. The screenshot below on the left shows the drop-down menu where Case Details is located; the screenshot on the right shows a sample case where Case ID is XX10008-00.



## Start Interview

To start an interview, FIs can touch and hold a Case ID on the case grid screen and select Start Interview from the drop-down menu that appears. This will launch the interview, beginning with the introduction. Buttons at the bottom of the screen will allow the FI to back up to the previous question, break off the interview, fast forward to another question, or advance to the next screen.

The interview consists of a series of questions, and at the end of the interview, event codes may be assigned based on the results of the interview. The interview data and all the CMS data are stored in SQLite databases on the data collection device. In the GATS project, the database file is named cmsdb.db3 file. It is typically stored in a folder on the handheld device.



## View Visit Record

As FIs perform field work on a case, they can enter event codes (for a given case) that document the work performed. The system allows FIs to enter event codes from a list of available events. The events for a given project are table driven and reasonably easy to adapt from project to project, although rules or validations based on event codes may require CMS code changes to implement. The definitions of pending and complete are controlled by parameters specified in the CMS setup INI table that lists the event codes associated with a given state. (See *Chapter 6* for information about the INI table in the CMS database [CMSDB].) An FI can also enter comments and specify the date and time of an event. A sample Visit Record screen is shown below for a case where three visits to the household have been made on three different dates.

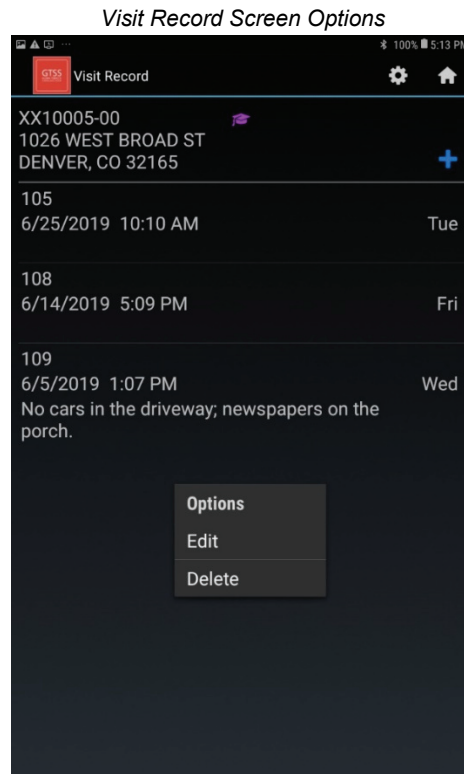
An FI can add an event by tapping the blue plus symbol on the Visit Record screen. When an FI enters an event code by tapping the menu item to select an event, the system goes through a series of checks to validate the event. If the event fails to validate, the system prompts the FI to take appropriate actions. The major validations are as follows:

- Some events require predecessors with certain conditions. If the conditions are not met, such an event cannot be accepted. The specific rules are updated for a given project.
- Some events require authorization codes. If the code is missing, such an event cannot be accepted.
- Some events will trigger other events, which most often are handled automatically in the system.

These rules or validations may require CMS code changes from project to project.

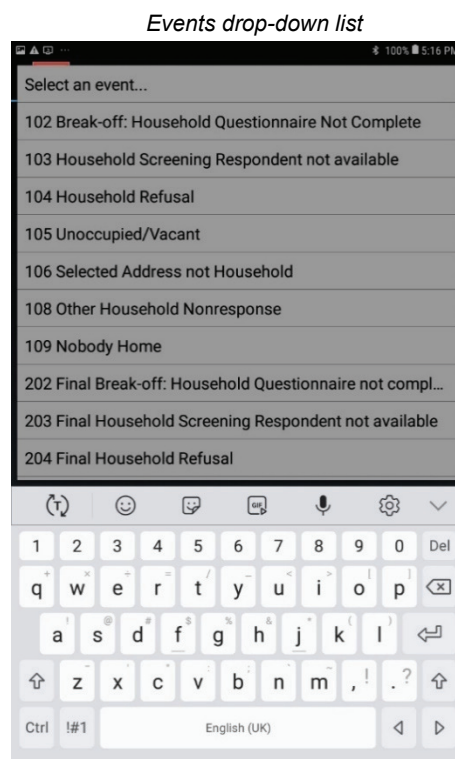
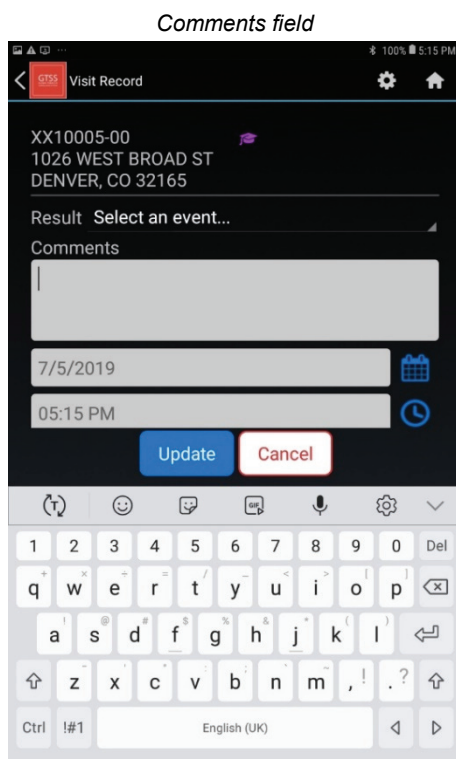


If an FI edits or deletes an event, the system reverses some of the steps described above. After an FI enters an event, and before the event is transmitted to the central servers, the FI has the option to edit or delete the event. Touch and hold an event, which will cause a menu to pop up offering the options Edit or Delete.



Once an event has been sent to the central servers, it should no longer be edited. See the *GATS Field Interviewer Manual* and *GATS Field Supervisor Manual* for details about GATS event use and protocol. The following two screenshots show the screen where the FI enters an event and comments (the Edit choice from the menu in the screenshot above). The FI selects a result or event code from a specific drop-down list. The FI can also enter a text comment, set the time and date, or accept the system defaults (current system date and time). The events are selected from the drop-down box labeled Result located above the Comments field. Part of the drop-down list available to GATS FIs is visible in the second screenshot.

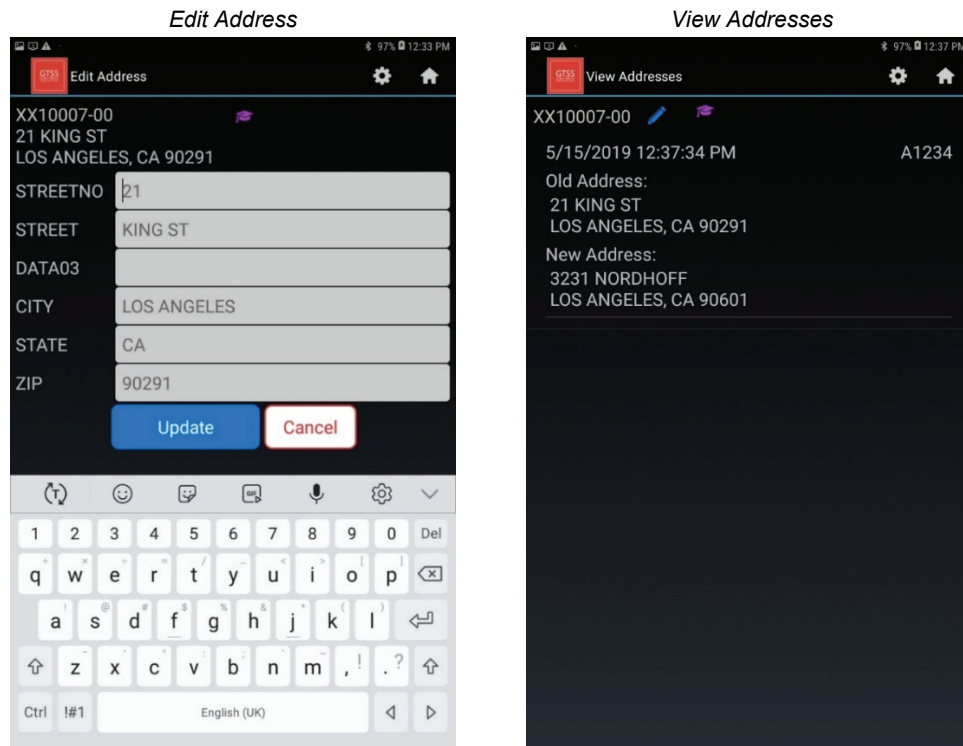




## **Edit Address**

The Edit Address option is a system parameter that may be turned on or off for the survey. If the option is turned on, an FI can edit the address using the menu item, Edit Address. Labels for these fields (DATA01, DATA02, etc.) can be renamed by editing the GSS messages table or the Questionnaire Designer (see *Chapter 4*) to be more appropriate for a given country's locator standards. Any address changes for a household require the transmission of an authorization code to the central servers. The default in GATS is to provide a generic authorization code so that FIs do not have to obtain one for changes. A sample Edit Address screen is shown in the following screenshot on the left.

This screen has an Update and a Cancel option that the FI can use to submit changes or exit from the screen with no changes.



### ***View Address Changes***

Users can view changes to an address for a given case by using the View Address Changes menu option.

**Step 1:** Tap and hold a specific case in the case grid to view the drop-down menu.

**Step 2:** Select View Address Changes.

The screenshot above right shows both the old and new addresses for the selected case, reflecting the changes that have been made. If no changes have been made, it will indicate that no changes have been made.

### ***Add a Case Note***

The Case Notes menu item allows FIs to maintain notes about a case in a diary, free text form. These notes are saved in the database in the Case Notes table and transmitted with other data. To make a case note or update a case note, follow these steps:

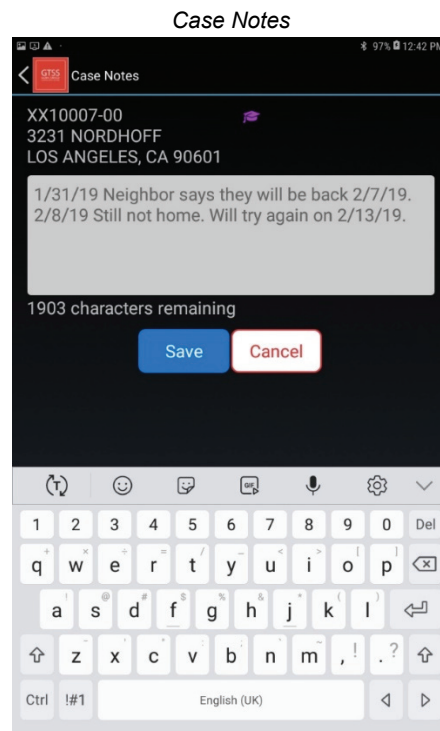
**Step 1:** Tap and hold a specific case in the case grid to view the drop-down menu.

**Step 2:** Select Add a Case Note in the drop-down menu to bring up the Case Notes screen.

**Step 3:** Enter notes in the text field available.

**Step 4:** Click "Save" to save the notes or cancel to exit from the screen.

The following screenshot shows an example of a note made for a selected case.



### 2.1.2 CMS Options

The next sections describe CMS action items that operate over cases. When executed, these action items will affect more than one case, report on multiple cases, set system-wide parameters, or change some system-wide setting. These action items are accessed from the menu at the top of the case management grid.

The following action items are available.

- Transmission — Allows the user to transmit a case or view the transmission log.
- Information — Provides access to the Version Report and Summary Statistics.
- Sort — Allows the user to rearrange the sort order of cases by Case ID, Number Case ID, Event Code, ZIP Code, City, or County.
- Data — Allows maintenance activities, including shrinking the database, checking the database, and exporting data.
- System settings — Changing system settings is an administrative function that requires an authorization code.
- More options — Allows the admin to select to Load Cases, Reload Training Cases, or Erase Training Cases.

Each of these choices is described below.

## Transmission

This option allows the user to transmit data, view the log, or test the connection. On the Transmission screen, tap **Transmit** to begin transmission of data to RTI, tap **Show Log** to view a history of previous data transmissions, or test the connection by using the **Test Conn** button. ("NO UPLOADS YET" indicates that no data has been transmitted.)

Field Interviewers may also use the FTP or Secure FTP option to send data to a project-operated FTP server. When the FI chooses this option, the GSS SQLite database file (CMSDB.DB3) is uploaded to the project server. It is uploaded to the FTP server specified in the FTP uniform resource locator (URL) setting and to the folder specified in the FTP Remote Folder setting. The file is renamed with the FIID and the current date, for example, 999001\_2019\_08\_23.db3, where 999001 is the FIID and 2019\_08\_23 is the date August 23, 2019. These files can then be input into the GSS aggregation software to start the process of building merged files.

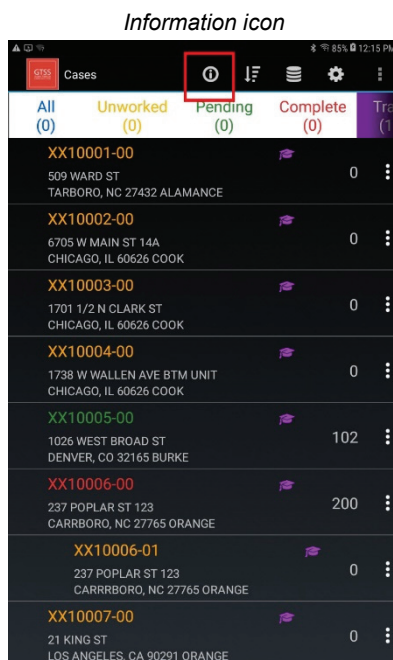
## Information Tab

The Information tab allows the user to see the Version Reporter and Summary Statistics.

The Version Reporter displays the current device settings, and other user- and device-related information, including:

- User Information—The Field Interviewer's ID and the Field Interviewer's name
- System Information—The current level of the battery
- Version information

The following screenshot shows the location of the Information icon.



Counts and percentages in the survey information section are shown for both the *Household Questionnaire* and the *Individual Questionnaire*. The rows are as follows:

- All—a total count of all the cases for that form.
- Unworked—a count and % of the number of cases that have not been touched (are at result code 0).
- Pending—a count and % of cases in the GATS pending status.
- Final—a count and % of cases in the GATS final status.
- Completed Interviews—a count and % of cases that are completed interviews.

An example of the Version Reporter and Summary Statistics screen is shown in the following screenshot.

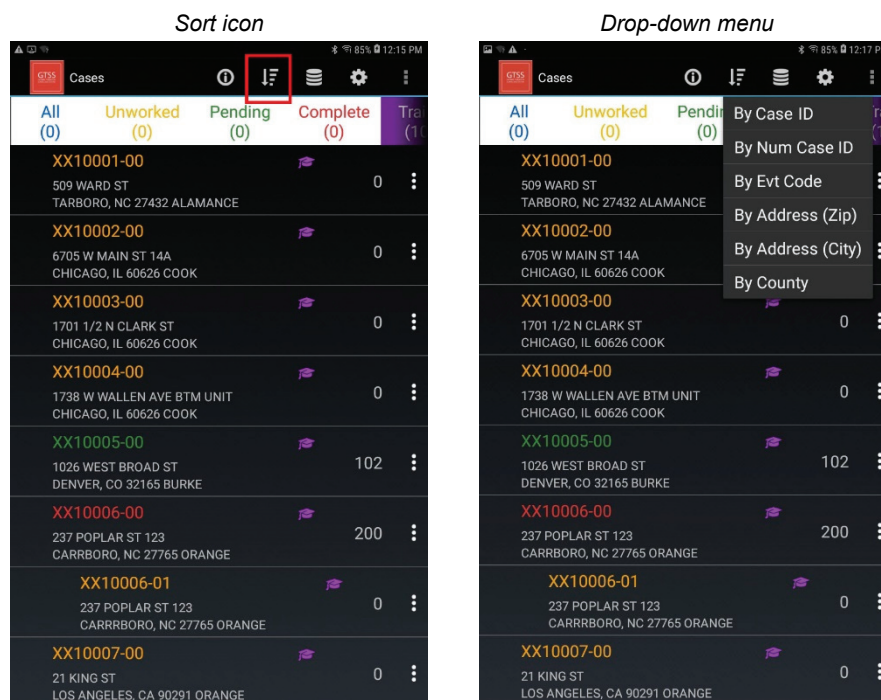
*Version Reporter and Summary Statistics*

USER INFORMATION		
FIID:	999998	
FI Name:	Jane Field	
SYSTEM INFORMATION		
Battery %:	88%	
System Date:	Friday, December 14 2018	
VERSION INFORMATION		
CMS INI:	Ver 10.3 12Nov2019	
Package:	CMS	
Version	3.1.20191112.0	
Version Code	14	
Date	Fri Dec 14 08:36:03 EST 2018	
SURVEY INFORMATION		
# Training Cases:	20	
# Live Cases:	0	
# Surveys	2	
Survey 0		
Ver.: 2019 09 11 094		
Total	10	
Unworked	6	60.0%
Pending	2	20.0%
Final	2	20.0%
Completed Interviews	2	20.0%

### Sort Menu

The Sort menu items allow users to call up different views of the data and displays of additional information. The View items (All, Unworked, Pending, Complete and Training) control which rows are displayed in the case grid, and the Sort items control the sort order of the selected rows. The user selects the view by tapping the specific option button. Data are not added or deleted using these options; they control only which and in what order data are displayed in the grid. You can control the view to show all cases, only pending cases, or only completed cases.

The following screenshots show the Sort icon and its drop-down menu.



The definitions of pending and completed are INI parameters that users can specify. The GATS definitions are shown below:

Variable Name	Variable Value	Comment
EvtCodeComplete	200, 400	Event codes for a completed questionnaire
EvtCodeFinal	200, 201, 202, 203, 204, 205, 206, 208, 209, 999, 400, 402, 403, 404, 407, 408, 409	Event codes that are final status codes
EvtCodePending	0, 102, 103, 104, 105, 106, 108, 109, 302, 303, 304, 307, 308, 309, 887	Event codes that are pending status codes
EvtCodeScreened	200	Event code indicating screening complete and one person selected for additional data collection

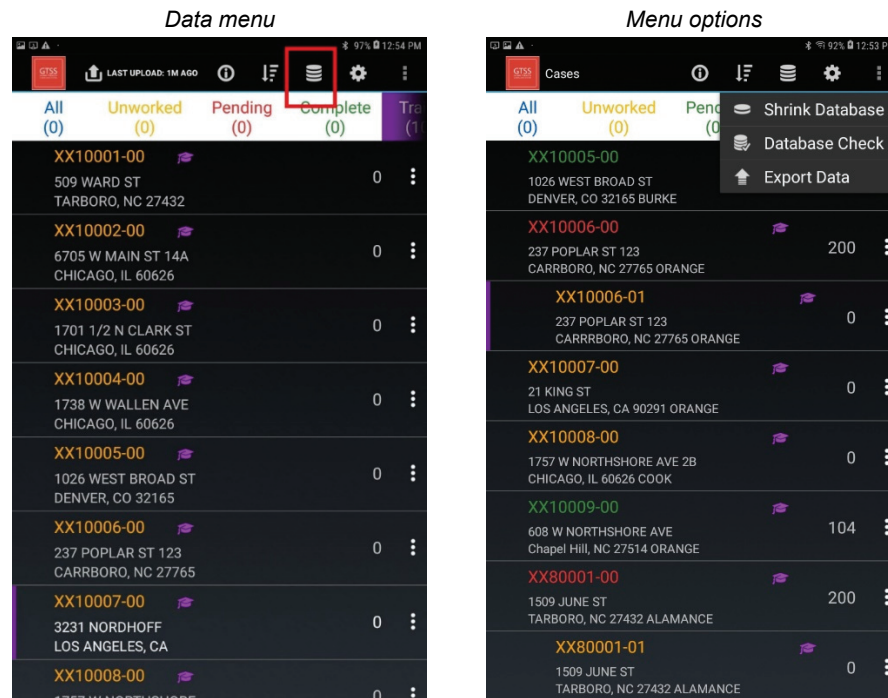
The Sort menu allows the users to control the sort order of the cases in the Select Screen. Cases can be sorted by a number of case variables (e.g., Case ID, Street, County, ZIP/PIN). The default sort order is by Case ID.

These actions operate on all cases currently loaded in the Android device.

### Data Menu

The Data menu provides access to the functions Shrink the Database, Database Check, and Export Data. Tap Data to view the drop-down menu as seen in the following screenshots:



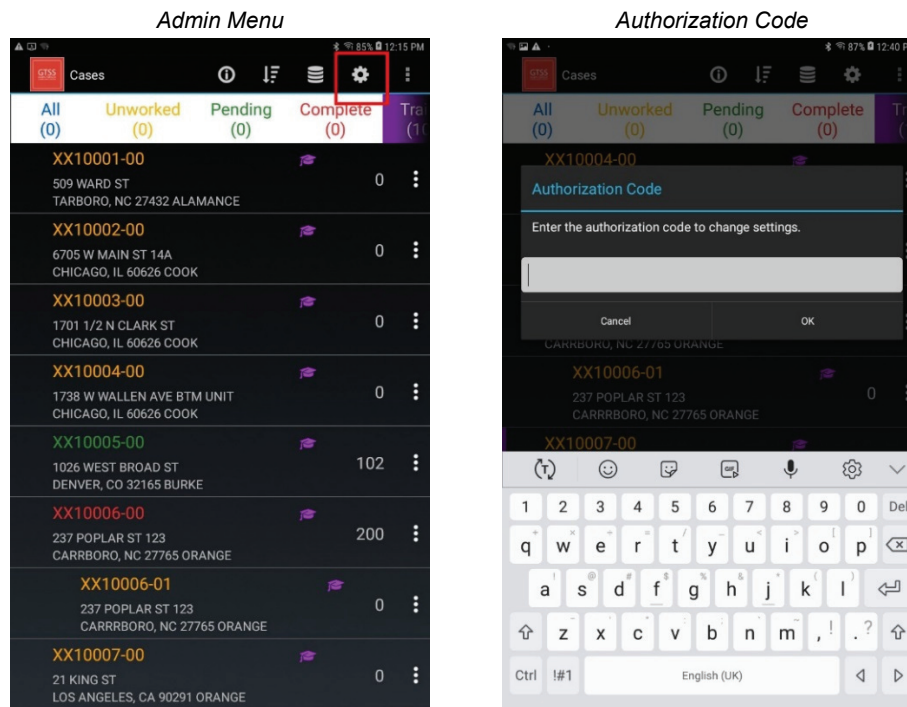


- The **Shrink Database** menu option allows maintenance on the internal database. It instructs the device to clean and compress the CMSDB database.
- The **Database Check** menu option provides a recap of cases, that is, number of cases, events, sent events, unsent events, responses, sent responses, unsent responses, media, sent media and unsent media.
- The CMS supports moving data into and out of the Android device via memory cards or other attachable devices that simulate disk drives in the OS. The memory cards can be used to make backups, load initial cases, update cases, remove or inactivate cases (to allow transfer to a different FI), and export data for aggregation across FIs. The **Export Data** action item makes a copy of the GSS database and copies it to the external memory device. The export file is copied to the `\\.\exports\` folder. This is the default folder. It can be changed by the user under the Admin Settings menu. Once the export is made, the export file can be copied to an external device and the export files used to aggregate or back up data in a number of ways. The copy of the GSS database is renamed to contain the Field Interviewer ID (FIID) and the current date. The name format is `XXXXXX_YYYY_MM_DD.db3`, where `XXXXXX` is the FIID, `YYYY` the year, `MM`, the month, and `DD` the day. If multiple exports are completed within the same day, the older file for the same day is overwritten.

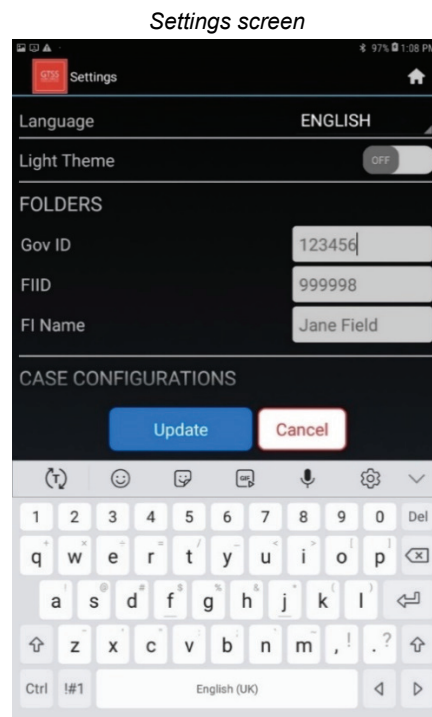
## Settings

When a user receives a data collection device, the machine must be configured by providing information such as the machine serial number, FIID, FI Name, FTP User Name, and FTP Password. In addition, the URL/path to the FTP server and export file destination folder of the FTP server for data transmissions can be specified. Until the system is configured, users cannot perform any other functions in the system. These items may be managed through the Admin Menu item Settings (see screenshot below). An

authorization code is required to change settings. This code may be retrieved from the GSS Tool Kit's Help menu by selecting Day Codes in the drop-down menu.



The Settings screen appears after the Day Code is provided. Scroll down the screen to see additional configuration settings.



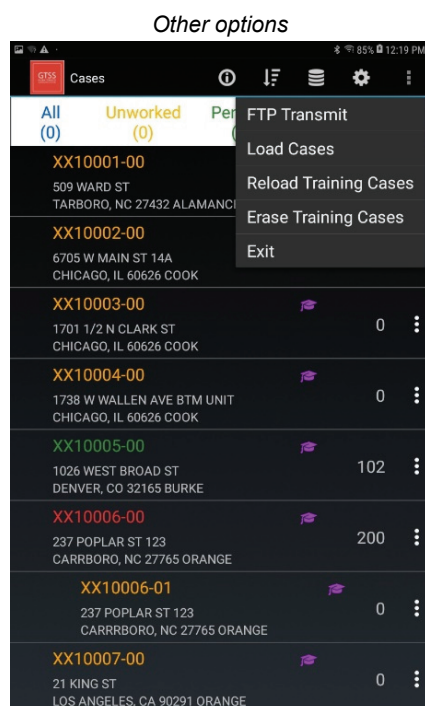


## Other Options

The drop-down menu for Other Options offers these selections:

- FTP Transmit
- Load Cases
- Reload Training Cases
- Erase Training Cases
- Exit

The following screenshot shows the drop-down menu.



- **FTP Transmit:** Field Interviewers can use the File Transfer Protocol (FTP) or Secure FTP to send and receive data to or from a project operated FTP server. When the FI chooses this option the GSS SQLite database file (CMSDB.DB3) is uploaded to the project server. It is uploaded to the FTP server specified in the FTP URL setting and to the folder specified in the FTP Remote Folder setting. The file is renamed with the FIID and the current date, for example 999001\_2019\_03\_23.db3, where 999001 is the FIID and 2019\_03\_23 is the date March 23, 2019. These files can then be input into the GSS aggregation software to start the process of building merged files.
- The **Load Cases** feature prompts the CMS to look for a file in the \...\exports\ folder. This is the default folder. It can be changed by the user under the Admin Settings menu. The Case File is named CaseFile.txt. This is the standard text file. If this file exists, the Load Cases feature will read it and load cases targeted for a given FI into the CMS case grid. The key for loading cases is the FIID, which is the FIID displayed in the Set Name and ID menu. That is, all cases in

CaseFile.txt with the same FIID as defined by the Set Name and ID screen item FIID will be loaded; all others will be ignored. After cases are loaded, the Case File will be renamed and saved in a backup location.

- Sample or mock cases available for training are stored in the default GSS database. FIs can load these cases by using the **Reload Training Cases** menu option. Training cases can be retrieved or removed at any time. Once loaded, a training case can be treated as a normal case, thus allowing practice use of the system, including transmission.
- If training cases are erased using the **Erase Training Case** menu option, all associated data are also deleted. Training cases have restricted ID numbers that start with XX to differentiate them from live cases. No live IDs that start with XX should ever be created. Data managers should be prepared to see and ignore cases that have Case IDs that start with XX in the GSS data tables. Users can edit the training case table (TrainingCases in the CMSDB file) to make the training cases look like live cases in their environment).
- The **Exit** choice will log the user out of the system.

### 3. GSS Engine

---

The GSS Engine is a specific module that displays questions on the device and collects data. The GSS Engine system was designed to allow most, if not all, of the specifications for a data collection task to be done by simply describing the attributes of the data to be collected and specifying the flow of data collection steps in a single metadata repository (a SQLite database and its data tables). The following sections explain how to create the tables that are required for the GSS Engine to generate a questionnaire and collect the data needed. A data collection activity can be as simple as an FI standing in a shopping center asking each passerby three or four questions—or it can be as involved as lengthy face-to-face interviews in a person's home or office. It can also be an inventory task or check-in/check-out process. In GATS, data collection is based on two questionnaires: The *Household Questionnaire* is designed to select an individual to interview in more detail in the *Individual Questionnaire*. The information needed to define a data collection activity is:

- the exact specification of a set of questions to ask,
- the exact specification of the answers that are allowed,
- the exact specification of the sequence or order in which to ask the questions, given the state of the system at any given point in the data collection, and
- the instructions to the data collector or respondent.

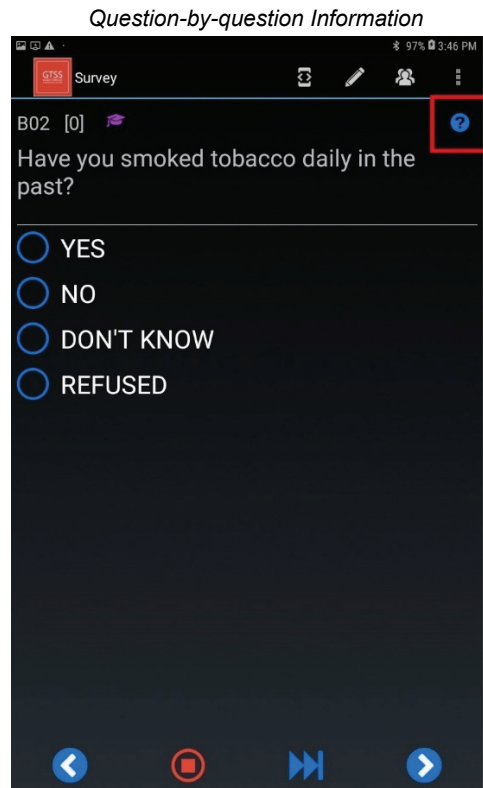
The GSS Engine considers a questionnaire to be a sequence of data collection screens. The screen is the building block that the system uses to create questionnaires. A screen is the display that the FI sees. It can contain instructions, text to read to the respondent, a field or fields that call for data input, navigation controls, menus, and status information. A questionnaire is then defined by tying together a sequence of screens to collect the information desired. In most cases, a screen collects one piece of data, for example, a respondent's first name, age, or response to a question. The GSS Engine treats all data collected from a single screen as a unit. The data from a screen are saved as a row in a permanent SQLite data table on the device and serve as the source to build analysis files of the collected data.

#### 3.1 Overall Design

The general design of the GSS Engine is based on the following key concepts:

- The unit of data collection is the screen.
- A screen has the following basic components:
  - A field for text (question or instructions),
  - One or more fields for data input (may contain lists to select from or blank fields to enter text or numbers), and
  - A set of buttons to control navigation among the screens (Back, Breakoff, Fast-forward, and Next).
- A set of icons that includes various options:
  - A Symbol Table that allows FI to view all data collected from previous screens;
  - Case Notes that allows FI to enter free-form information about the case;

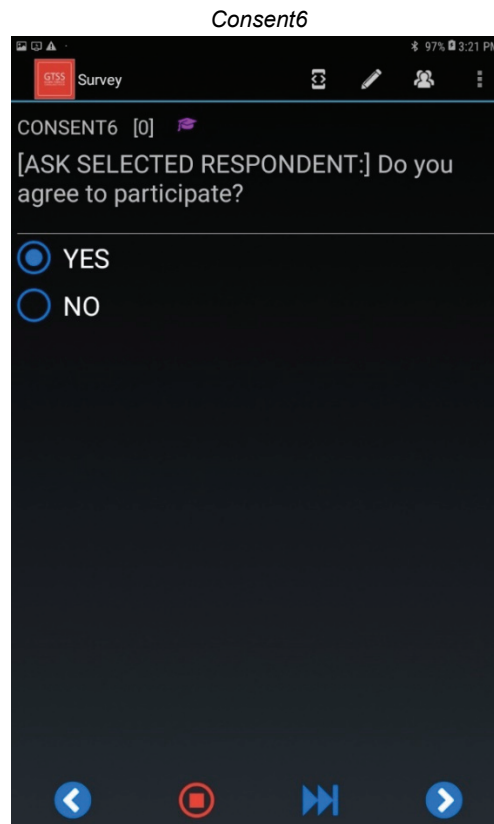
- A Roster that provides information on the values within a loop, for example, household resident details; and
  - Language selection allows users to select among the languages for the questionnaire (up to 20).
- A question mark icon that can provide question-by-question (QXQ) information about the question on a given screen.



- Movement forward and backward between the screens is controlled by standard Android gestures. A swipe from right to left (like turning a page in a book) moves you forward; we will call this a Forward swipe. A swipe from left to right moves you back a screen; we will call this a Back swipe. The navigation buttons may also be used.
- Each screen generates one data row in an output data table of response data (the Responses table) when the user moves to the next question.
- The data row captures the data item(s) from the screen plus control data (time stamp, language used, validity flag, etc.) each time the user moves to the next screen.
- Some screens do not generate any output data and/or may not be visible to the user (for example, Info screens or screens that invoke internal computations or control looping).

A questionnaire is developed by creating a collection of these screens, providing the metadata for each screen, and defining the sequence in which they will be shown to the data collector or data provider. The following screenshot (Consent6) displays a sample GSS Engine List type screen. In this type of screen, the FI reads the question text (for this example, "Do you agree to participate?") and then, depending upon

the answer received from the respondent, the FI taps the correct option choice (here, either YES or NO). In this screen the text in capital letters and within the square brackets "[ASK SELECTED RESPONDENT:]" is by GATS convention an instruction for the interviewer and is not read to the Respondent.



To go on to the next screen, the FI would perform a Forward swipe or choose the Forward icon. If the FI wanted to back up to a previous screen or go even further back, then the FI would use Back swipes to back up to as many screens as desired. Each time a screen is backed over, the data in the Responses table for the previous answer for that question is flagged as invalid. The elements of the screen are described in *Exhibit 3-1*.

**Exhibit 3-1. GSS Engine Screen Elements**

Item(s)	Element Name	Value Shown on Screen Consent6
Screen name	QID	Consent6
Nesting or loop level	Instance	[0]
Question text	QText	[ASK SELECTED RESPONDENT ...
Answer text choices	Atext(i)	YES, NO
Menu Items	Tools	Tools
	Language	Language
	Help QxQ	Help QxQ

The GSS Engine supports a number of screen types that the questionnaire developer can use to build the data collection application. *Exhibit 3-2* lists the screen types that are supported. They fall into three categories:

- **Data capture:** List, All That Apply (ATA), Numeric (Num), Text, Date, GPS capture, Photo, BarCode.
- **Data display:** Info.
- **Navigation or computation:** Start, End, Loop Start (Loop), Loop End (Lend), Compute (Comp).

## 3.2 Example 1—Programming a Small Questionnaire

**Exhibit 3-2. GSS Engine Screen Types**

Screen Type ID	Screen Description	Generates Data Row	# of Input Fields	Type Data Collected
List	Select one item from a fixed list of choices	Yes	1	Code associated with the selected choice
ATA	Select one or more items from a fixed list of choices	Yes	Up to 31	Single numeric code based on which mix of items selected 0 to 2 <sup>31</sup>
Num	Enter a single numeric item	Yes	1	Numeric; can be floating point or restricted to integer
Text	Enter a single text item	Yes	1	Character string, maximum length 32,767; only first 255 will be stored
Date	Enter a single date item containing month, day, year	Yes	1	Date stored as character string in MM/DD/YYYY format
Info	Presents only text information to user; does not collect anything	Yes	0	No data collected on this screen
Start	Required screen—must be the first screen executed	No	0	No data collected
End	Required screen—must be the last screen executed	No	0	No data collected
Loop	Not visible to user	No	0	Used to define a loop start
Lend	Not visible to user	No	0	Used to define a loop end
Comp	Not visible to user	No	0	Used to implement user logic calls
Photo	Captures an image using the device camera	No	0	Image saved as a file with name as XXX.YYY.zzz.jpg
GPS	Captures location data using the device GPS or other locator info	Yes	1	Captures GPS data and stores it as a single character string
BarCode	Reads a barcode using the device camera	Yes	1	Stores the barcode in the answer field, also stores result in symbol table

An example of a small questionnaire may help to show how a developer might build a data collection questionnaire or form. This example is based on a few simple questions from the GATS *Individual Questionnaire*. The GSS programming specifications for these questions are shown in *Exhibit 3-3*.

### Exhibit 3-3. Individual Questionnaire Subset

Question Name	Question Attributes	Next Question																																													
START	Ver.: 2019.04.17.043 <span style="float: right;">Comp</span>	CONSENT1																																													
CONSENT1	<p style="text-align: right;">CONSENTAGE LIST</p> <p>[SELECT THE APPROPRIATE AGE CATEGORY BELOW. IF NEEDED, CHECK THE AGE OF SELECTED RESPONDENT FROM THE "CASE INFO" SCREEN IN THE TOOLS MENU.]</p> <table border="1"> <thead> <tr> <th>Seq</th><th>Code</th><th>English</th></tr> </thead> <tbody> <tr> <td>1</td><td>1</td><td>15-17</td></tr> <tr> <td>2</td><td>2</td><td>18 OR OLDER</td></tr> <tr> <td>3</td><td>3</td><td>EMANCIPATED MINOR (15-17)</td></tr> </tbody> </table>	Seq	Code	English	1	1	15-17	2	2	18 OR OLDER	3	3	EMANCIPATED MINOR (15-17)	CONSENT2																																	
Seq	Code	English																																													
1	1	15-17																																													
2	2	18 OR OLDER																																													
3	3	EMANCIPATED MINOR (15-17)																																													
CONSENT2	<p style="text-align: right;">INFO</p> <p>Before starting the interview, I need to obtain consent from a parent or guardian of [NAME OF RESPONDENT] and from [NAME OF RESPONDENT]. &lt;P&gt;  [IF BOTH SELECTED RESPONDENT AND PARENT/GUARDIAN ARE AVAILABLE, CONTINUE WITH INTERVIEW. &lt;P&gt;  IF PARENT/GUARDIAN IS NOT AVAILABLE, BREAK-OFF INTERVIEW AND SCHEDULE AN APPOINTMENT TO RETURN. &lt;P&gt;  IF MINOR RESPONDENT IS NOT AVAILABLE, CONTINUE WITH OBTAINING PARENTAL CONSENT.]</p>	A02A																																													
A02A	<p style="text-align: right;">MONTHLIST LIST</p> <p>What is the month of your date of birth?</p> <table border="1"> <thead> <tr> <th>Seq</th><th>Code</th><th>English</th></tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>01</td></tr> <tr><td>2</td><td>2</td><td>02</td></tr> <tr><td>3</td><td>3</td><td>03</td></tr> <tr><td>4</td><td>4</td><td>04</td></tr> <tr><td>5</td><td>5</td><td>05</td></tr> <tr><td>6</td><td>6</td><td>06</td></tr> <tr><td>7</td><td>7</td><td>07</td></tr> <tr><td>8</td><td>8</td><td>08</td></tr> <tr><td>9</td><td>9</td><td>09</td></tr> <tr><td>10</td><td>10</td><td>10</td></tr> <tr><td>11</td><td>11</td><td>11</td></tr> <tr><td>12</td><td>12</td><td>12</td></tr> <tr><td>13</td><td>-7</td><td>DON'T KNOW</td></tr> <tr><td>14</td><td>-9</td><td>REFUSED</td></tr> </tbody> </table>	Seq	Code	English	1	1	01	2	2	02	3	3	03	4	4	04	5	5	05	6	6	06	7	7	07	8	8	08	9	9	09	10	10	10	11	11	11	12	12	12	13	-7	DON'T KNOW	14	-9	REFUSED	A02B
Seq	Code	English																																													
1	1	01																																													
2	2	02																																													
3	3	03																																													
4	4	04																																													
5	5	05																																													
6	6	06																																													
7	7	07																																													
8	8	08																																													
9	9	09																																													
10	10	10																																													
11	11	11																																													
12	12	12																																													
13	-7	DON'T KNOW																																													
14	-9	REFUSED																																													

Question Name	Question Attributes	Next Question																																	
A02B	<p style="text-align: right;">NUM</p> <p>What is the year of your date of birth?</p> <p>[Range: 1905 - 2005]</p> <p>if {A02a} == -7    {A02b} == -7    {A02a} == -9    {A02b} == -9 then goto A03;</p> <p>NoDKRE,Integer,RngInclude = -7;-9;</p>	A04																																	
A03	<p style="text-align: right;">NUM</p> <p>How old are you?&lt;P&gt; [IF RESPONDENT IS UNSURE, PROBE FOR AN ESTIMATE AND RECORD AN ANSWER&lt;P&gt; IF REFUSED, BREAK-OFF AS WE CANNOT CONTINUE INTERVIEW WITHOUT AGE]</p> <p>[Range: 5 - 110]</p> <p>NoDKRE,Integer</p>	A04																																	
A04	<p style="text-align: right;">EDUCATION LIST</p> <p>What is the highest level of education you have completed?&lt;P&gt;&lt;P&gt; [SELECT ONLY ONE CATEGORY]&lt;P&gt;&lt;P&gt; [ADJUST CATEGORIES FOR SPECIFIC COUNTRY]</p> <table border="1"> <thead> <tr> <th>Seq</th><th>Code</th><th>English</th></tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>NO FORMAL SCHOOLING</td></tr> <tr><td>2</td><td>2</td><td>LESS THAN PRIMARY SCHOOL COMPLETED</td></tr> <tr><td>3</td><td>3</td><td>PRIMARY SCHOOL COMPLETED</td></tr> <tr><td>4</td><td>4</td><td>LESS THAN SECONDARY SCHOOL COMPLETED</td></tr> <tr><td>5</td><td>5</td><td>SECONDARY SCHOOL COMPLETED</td></tr> <tr><td>6</td><td>6</td><td>HIGH SCHOOL COMPLETED</td></tr> <tr><td>7</td><td>7</td><td>COLLEGE/UNIVERSITY COMPLETED</td></tr> <tr><td>8</td><td>8</td><td>POST GRADUATE DEGREE COMPLETED</td></tr> <tr><td>9</td><td>-7</td><td>DON'T KNOW</td></tr> <tr><td>10</td><td>-9</td><td>REFUSED</td></tr> </tbody> </table>	Seq	Code	English	1	1	NO FORMAL SCHOOLING	2	2	LESS THAN PRIMARY SCHOOL COMPLETED	3	3	PRIMARY SCHOOL COMPLETED	4	4	LESS THAN SECONDARY SCHOOL COMPLETED	5	5	SECONDARY SCHOOL COMPLETED	6	6	HIGH SCHOOL COMPLETED	7	7	COLLEGE/UNIVERSITY COMPLETED	8	8	POST GRADUATE DEGREE COMPLETED	9	-7	DON'T KNOW	10	-9	REFUSED	100
Seq	Code	English																																	
1	1	NO FORMAL SCHOOLING																																	
2	2	LESS THAN PRIMARY SCHOOL COMPLETED																																	
3	3	PRIMARY SCHOOL COMPLETED																																	
4	4	LESS THAN SECONDARY SCHOOL COMPLETED																																	
5	5	SECONDARY SCHOOL COMPLETED																																	
6	6	HIGH SCHOOL COMPLETED																																	
7	7	COLLEGE/UNIVERSITY COMPLETED																																	
8	8	POST GRADUATE DEGREE COMPLETED																																	
9	-7	DON'T KNOW																																	
10	-9	REFUSED																																	
100	<p style="text-align: right;">INFO</p> <p>Those are all of the questions I have. Thank you very much for participating in this important survey.&lt;P&gt; [YOU WILL NOT BE PERMITTED TO BACK UP FROM THE NEXT SCREEN.]</p>	END																																	



To develop these specifications as a GSS application, you can use eight screens. The screen names, order of presentation, and screen types are shown below:

	<b>Question Name</b>	<b>Screen Type</b>
1.	START	Info
2.	CONSENT1	Info
3.	CONSENT2	Info
4.	A02A	List
5.	A02B	Num
6.	A03	Num
7.	A04	List
8.	I00	Info

No loops or routing paths are used in the very simple example above. Only three different GSS screen types are required for the application (Info, List, and Num). To build the screens needed for this example, place the information in SQLite tables. *Exhibits 3-4*, *3-5*, *3-6*, and *3-7* show the four SQLite tables (Questions, Texts, Answers, and INI, respectively) required to build this example. Of these tables, the INI table is a repository for general parameters for the whole questionnaire and GSS setup for its execution. The other three tables contain the question-level information that defines the questionnaire.

The Questions table contains the information that pertains to the question, and it has links to the two other tables for additional information. The Texts table is linked by the variable Question ID (QID) to the Questions table and contains the text for a given question. In the Texts table you must provide one row for each QID for each language you wish to support. If you plan to use English, Spanish, and French, then every QID with text needs three rows. The Answers table is somewhat similar in that it contains the information that defines the List and All that Apply answer choices. Again, these lists can be in as many languages as needed. Each Answer Set is repeated for every language required. In *Exhibit 3-4* (the Questions table), each row represents the metadata for a given question/screen. Every GSS questionnaire must begin with a Start screen (QSequence 1 in *Exhibit 3-4*) and end with an End screen (QSequence 271 in *Exhibit 3-4*). The first two columns provide ID information for the system: a sequence number that provides the default order for listing the questions and a name for the screen (QID).

**Exhibit 3-4. Screen Information: Questions Table**

QSequence	Qid	QNext	Qlogic	QType	Format	Qanswer	Range Lo	Range Hi	Loop Start	Loop End	Loop GoTo	Roster #	Roster Col
1	START	CONSENT1		COMP		None	0	0			0	0	0
21	CONSENT1	CONSENT2		LIST	PASSIVE_GPS_CAPTURE;	CONSENTAGE	0	0			0	0	0
25	INTRO_PASSIVE_GPS	END		TEXT		None	0	0			0	0	0
31	CONSENT2	A02A		INFO		None	0	0			0	0	0
41	A02A	A02B		LIST		MONTHLIST	0	0			0	0	0
51	A02B	A04	if {A02a} == -7    {A02b} == -7    {A02a} == -9    {A02b} == -9 then goto A03;	NUM	integer;NODKRE, RngInclude=-7,-9	None	1905	2005			0	0	0
61	A03	A04		NUM	integer;NODKRE;	None	5	110			0	0	0
65	A04	I00		LIST		EDUCATION	0	0			0	0	0
81	I00	END		INFO		None	0	0			0	0	0
271	END	Null		END		None	0	0			0	0	0

**Exhibit 3-5. Question Text: Texts Table**

Pkey	QID	Language	Qtext
7647	START	ENGLISH	Ver.: 2019 04 17 042
7665	CONSENT1	ENGLISH	[SELECT THE APPROPRIATE AGE CATEGORY BELOW. IF NEEDED, CHECK THE AGE OF SELECTED RESPONDENT FROM THE "CASE INFO" SCREEN IN THE TOOLS MENU.]
7436	CONSENT2	ENGLISH	Before starting the interview, I need to obtain consent from a parent or guardian of [NAME OF RESPONDENT] and from [NAME OF RESPONDENT]. <P> [IF BOTH SELECTED RESPONDENT AND PARENT/GUARDIAN ARE AVAILABLE, CONTINUE WITH INTERVIEW. <P> IF PARENT/GUARDIAN IS NOT AVAILABLE, BREAK-OFF INTERVIEW AND SCHEDULE AN APPOINTMENT TO RETURN. <P> IF MINOR RESPONDENT IS NOT AVAILABLE, CONTINUE WITH OBTAINING PARENTAL CONSENT.]
7445	A02A	ENGLISH	What is the month of your date of birth?
7240	A02B	ENGLISH	What is the year of your date of birth?
7239	A03	ENGLISH	How old are you?<P> [IF RESPONDENT IS UNSURE, PROBE FOR AN ESTIMATE AND RECORD AN ANSWER<P> IF REFUSED, BREAK-OFF AS WE CANNOT CONTINUE INTERVIEW WITHOUT AGE]
7447	A04	ENGLISH	What is the highest level of education you have completed?<P><P> [SELECT ONLY ONE CATEGORY]<P><P> [ADJUST CATEGORIES FOR SPECIFIC COUNTRY]
5442	I00	ENGLISH	Those are all of the questions I have. Thank you very much for participating in this important survey.<P> [YOU WILL NOT BE PERMITTED TO BACK UP FROM THE NEXT SCREEN.]

**Exhibit 3-6. Answer Text and Codes: Answers Table**

UID	AID	Language	ASequence	ACode	AText
558	EDUCATION	ENGLISH	1	1	NO FORMAL SCHOOLING
559	EDUCATION	ENGLISH	2	2	LESS THAN PRIMARY SCHOOL COMPLETED
560	EDUCATION	ENGLISH	3	3	PRIMARY SCHOOL COMPLETED
561	EDUCATION	ENGLISH	4	4	LESS THAN SECONDARY SCHOOL COMPLETED
562	EDUCATION	ENGLISH	5	5	SECONDARY SCHOOL COMPLETED
563	EDUCATION	ENGLISH	6	6	HIGH SCHOOL COMPLETED
564	EDUCATION	ENGLISH	7	7	COLLEGE/UNIVERSITY COMPLETED
565	EDUCATION	ENGLISH	8	8	POST GRADUATE DEGREE COMPLETED
566	EDUCATION	ENGLISH	9	-7	DON'T KNOW
567	EDUCATION	ENGLISH	10	-9	REFUSED
568	CONSENTAGE	ENGLISH	1	1	15-17
569	CONSENTAGE	ENGLISH	2	2	18 OR OLDER
570	CONSENTAGE	ENGLISH	3	3	EMANCIPATED MINOR (15-17)
571	MONTHLIST	ENGLISH	1	1	01
572	MONTHLIST	ENGLISH	2	2	02
573	MONTHLIST	ENGLISH	3	3	03
574	MONTHLIST	ENGLISH	4	4	04
575	MONTHLIST	ENGLISH	5	5	05
576	MONTHLIST	ENGLISH	6	6	06
577	MONTHLIST	ENGLISH	7	7	07
578	MONTHLIST	ENGLISH	8	8	08
579	MONTHLIST	ENGLISH	9	9	09
580	MONTHLIST	ENGLISH	10	10	10
581	MONTHLIST	ENGLISH	11	11	11
582	MONTHLIST	ENGLISH	12	12	12
583	MONTHLIST	ENGLISH	13	-7	DON'T KNOW
584	MONTHLIST	ENGLISH	14	-9	REFUSED

**Exhibit 3-7. Initialization Data: INI Table for Survey Form**

VariableName	VariableValue	Comment
Breakoff	Yes	Flag to control breakoff option (Yes/No)
DisplayOldData	Yes	If old answers exist show them on back up (Yes/No)
FastForwardOK	Yes	Switch to turn on or off Fast Forward option: Yes or No
InternalCMS	No	Use the internal CMS system?
LoginRequired	No	For internal CMS Yes if want to force login
NumLanguages	1	Number of languages used
Qversion	Ver.4.0 May 2012	Version of the INI file
QxQOption	Yes	Turn on QxQ option: Yes or No
ROSTERLABEL1	Name Age B Mo B Yr Gender Smkr	Labels for the Roster Columns, 0....n
StartUpLanguage	0	index of language to use as startup
SurveyID	AndroidTest1	Unique ID for this Survey
TestingRndID	No	used to generate random IDs if = Yes
TextToCaps	Yes	convert all text entry to CAPS (Yes/No)
TimeZoneOffset	0	Pseudo Time Zone used by CMS to reset PPC
LangSpecs01	ENGLISH,Tahoma,09	Language # 1 and font specs
LangSpecs02	RUSSIAN,Tahoma,10	French
LangSpecs03	Spanish,Tahoma,11	
DisableMainMenu	Yes	Disable the main menu (Start Screen) if Yes. If no keep the main menu
FastForwardCount	50	Max number of questions to fast forward, there is a limit or GSS may crash

The next column in the Questions table, QNext, defines the question to go to next after you Forward swipe on the screen. In the QLogic column, you can enter GSS programming commands in a Java-like programming language. These logic statements allow you to create variables and alter program flow and output data. *Appendix A* discusses the QLogic programming language and how to code commands for this function in GSS.

In column QType, you specify the type of screen. The formats column allows you to provide some formatting directives to the system. (See *Exhibit 3-8* for a complete list of the available formatting options.) The formats column can be updated in the Special Instructions section of the questionnaire designer tool. In column QAnswer, you specify the name of the answer set (UID) in the survey database table Answers that corresponds to a given List or ATA screen. This is the name of a set of answers to be used as answer options. There must be a set (one or more) of these answers in the Answers table to match the reference in the Questions table. The next two columns, RangeLo and RangeHi, provide a place to specify a minimum and maximum range for numeric type screens. If the answer to a question falls outside the range specification ( $\text{RangeLo} \leq \text{Answer} \leq \text{RangeHi}$ ), it will generate validation errors and force you to input data within the range. It is also possible to specify ranges as a list.

**Exhibit 3-8. GSS Engine Special Instructions Options**

Option Name	Function	Applicable Screen Types	Parameters
Integer	Forces input to whole numbers only (no decimal points)	Num	None
NODK	Requires a response	Text, Num	None
RngInclude	Directive to include values into a range check on a numeric variable in addition to that specified by RangeHi and RangeLo	Num	For example, RngInclude = 99; 98; 97, this will add the values 99, 98, and 97 to the valid range
RngExcp	Directive to exclude values from a range check on a numeric variable	Num	For example, RngExcp = 2; 8, excludes 2 and 8 as legal values in a specified range
NoBackup	Restricts backup at this screen, does not allow Field Interviewer to back up over the screen	All	None
NoBreakOff	Restricts Break Off at this screen, does not allow Field Interviewer to use the Break Off menu option on this screen	All	None

*Exhibit 3-9* details the functions and requirements for each column of a Questions table. The question text is stored in a separate table, the Texts table (see *Exhibit 3-5*). The Texts table has a row for every QID and language. Since this example is English only, the Texts table has a row for each QID in the Questions table that has question text.

### Exhibit 3-9. GSS Screen Properties

Attribute Name	Function	Requirements	Notes
Sequence #	Controls the sort order of screens; not used for flow, just presentation of rows for viewing or printing	Number in the range 1.000 to 99999.999	
QID	Name of screen	Must be unique within questionnaire and <= 20 characters	
QType	Type of screen	See <i>Exhibit 3-2</i> for list of available types	
AnswerSet	Name of list of an answer set of response options for List or ATA type questions (<= 20 characters)	Verbatim text that will be displayed in List and ATA type screens	Can use the same answer set for multiple questions
Logic	GSS programming language for logic and flow control	See <i>Appendix A</i> for discussion of coding conventions for QLogic	
Special Instructions	List for formatting or other special options that apply to the given screen (see <i>Exhibit 3-8</i> )		
RangeLo	Range for QType Num used for a validity check	Numeric value	Low value can be any real number
RangeHi	Range for QType Num used for a validity check	Numeric value	High value can be a real number >= RangeLo
Loop Info	Parameters used to define loops		
LoopStart	Symbol or constant that defines starting loop counter value		
LoopEnd	Symbol or constant that defines ending loop counter value		
LoopGoto	Screen to go to when loop starts or ends		
Roster #	Roster # to store a copy of this answer	1, 2, 3, or 4	Typically used to build grids of data collected in loops for later editing or display
Roster Col	Column # of the roster to store a copy of the data	0 to 20	The row # is determined by the instance level of the questionnaire but user may choose column

The set of specifications provided previously leads to the following set of screens of the GSS shown below. Note that the last screen comes from the End screen defined in the Questions table.

Consent1

Consent2

A02A



Consent2

Consent2 [0]

Before starting the interview, I need to obtain consent from a parent or guardian of [NAME OF RESPONDENT] and from [NAME OF RESPONDENT].

[IF BOTH SELECTED RESPONDENT AND PARENT/GUARDIAN ARE AVAILABLE, CONTINUE WITH INTERVIEW.

IF PARENT/GUARDIAN IS NOT AVAILABLE, BREAK-OFF INTERVIEW AND SCHEDULE AN APPOINTMENT TO RETURN.

IF MINOR RESPONDENT IS NOT AVAILABLE, CONTINUE WITH OBTAINING PARENTAL CONSENT.]

A02A

A02a [0]

What is the month of your date of birth?

☐ 01

☐ 02

☐ 03

☐ 04

☐ 05

☐ 06

☐ 07

☐ 08

☐ 09

☐ 10

☐ 11

☐ 12

A02B

A03 [0]

How old are you?

[IF RESPONDENT IS UNSURE, PROBE FOR AN ESTIMATE AND RECORD AN ANSWER

IF REFUSED, BREAK-OFF AS WE CANNOT CONTINUE INTERVIEW WITHOUT AGE]

1 2 3

4 5 6

7 8 9 -

0

A03

A02b [0]

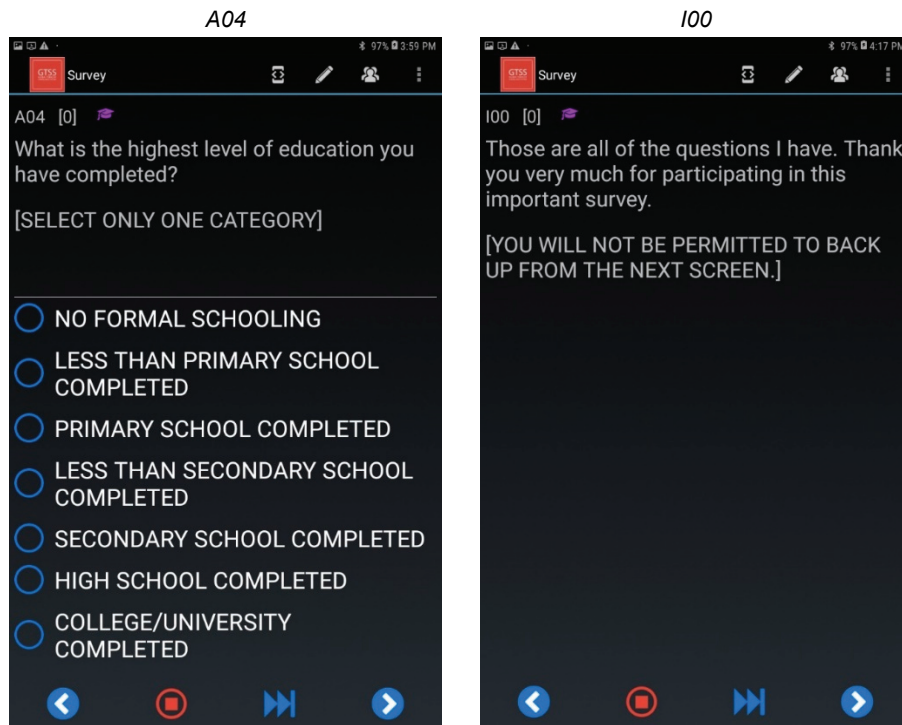
What is the year of your date of birth?

1 2 3

4 5 6

7 8 9 -

0



### 3.3 Example 2—Using GSS to Program the GATS Household Questionnaire

A second, more complicated example provides more details on GSS programming. This example is based on the screens similar to those used to implement the 2019 version of the GATS HQ. The GATS HQ requires collection of data on the members of a household and selection of one of those members to participate in a second interview. The GSS programming specifications of this example are shown in *Exhibit 3-10*.

### Exhibit 3-10. Household Questionnaire

Question Name	Question Attributes	Next Question
START	Ver.: 2020 05 04 099 <span>COMP</span>	INTRO
INTRO	<p><span>INFO</span></p> <p>[THE HOUSEHOLD SCREENING RESPONDENT SHOULD BE 18 YEARS OF AGE OR OLDER AND YOU MUST BE CONFIDENT THAT THIS PERSON CAN PROVIDE ACCURATE INFORMATION ABOUT ALL MEMBERS OF THE HOUSEHOLD. IF NEEDED, VERIFY THE AGE OF THE HOUSEHOLD SCREENING RESPONDENT TO MAKE SURE HE/SHE IS 18 YEARS OF AGE OR OLDER.&lt;P&gt;</p> <p>THE HOUSEHOLD SCREENING RESPONDENT CAN BE LESS THAN 18 YEARS OLD, ONLY IF NO HOUSEHOLD MEMBERS ARE 18 YEARS OF AGE OR OLDER.]</p> <p><span>PASSIVE_GPS_CAPTURE;</span></p>	INTRO1
INTRO_PASSIVE_GPS	<p><span>TEXT</span></p> <p>[THIS QUESTION IS NOT ASKED. IT IS USED TO STORE GPS LOCATION DATA]</p>	END
INTRO1	<p><span>INFO</span></p> <p>An important survey of adult tobacco use behavior is being conducted by the &lt;B&gt;&lt;I&gt;[FILL COUNTRY SPONSORING AGENCY]&lt;/I&gt;&lt;/B&gt; throughout &lt;B&gt;&lt;I&gt;[FILL COUNTRY]&lt;/I&gt;&lt;/B&gt; and your household has been selected to participate. All houses selected were chosen from a scientific sample and it is very important to the success of this project that each participates in the survey. All information gathered will be kept strictly confidential. I have a few questions to find out who in your household is eligible to participate.</p> <pre> set {HHType} = '{Parm2}'; set {HHTypeLowerCaseENG} = 'both'; if {HHType} == 'Male' then set {HHTypeLowerCaseENG} = 'male'; if {HHType} == 'Female' then set {HHTypeLowerCaseENG} = 'female'; set {HHTypeUpCaseENG} = 'BOTH'; if {HHType} == 'Male' then set {HHTypeUpCaseENG} = 'MALE'; if {HHType} == 'Female' then set {HHTypeUpCaseENG} = 'FEMALE'; </pre>	HH1

Question Name	Question Attributes	Next Question
HH1	<p style="text-align: right;">NUM</p> <p>First, I'd like to ask you a few questions about your household. In total, how many persons live in this household? &lt;P&gt;</p> <p>[INCLUDE ANYONE WHO CONSIDERS THIS HOUSEHOLD THEIR USUAL PLACE OF RESIDENCE]</p> <p>[Range: 0 - 50]</p> <p>if {HH1} == 0 then goto NoEligible;</p> <p>integer;NODKRE;</p>	HH2
HH2	<p style="text-align: right;">NUM</p> <p>How many of these household members are 15 years of age or older?</p> <p>[Range: 0 - 20]</p> <p>if {HHType} == 'Both' then set {HH3} = {HH2}; if {HH2} == 0 then goto NoEligible; If {HH2} &gt; {HH1} then goto TooMany; If {HH1} &gt; {HH2} then goto HH2A; if {HHType} == 'Both' then goto HH4both;</p> <p>integer;NODKRE;</p>	HH3
TOOMANY	<p style="text-align: right;">INFO</p> <p>[YOU CAN'T/SHOULDN'T HAVE MORE PEOPLE GREATER THAN OR EQUAL TO 15 YEARS OLD THAN THERE ARE TOTAL HH MEMBERS; PLEASE DOUBLE CHECK THE ANSWERS SO FAR.]</p> <p>backupto HH2;</p>	HH2
HH2A	<p style="text-align: right;">NUM</p> <p>How many household members are less than 5 years old?</p> <p>[Range: 0 - 20]</p> <p>SET {HHTOTAL} = {HH2} + {HH2A}; If {HHTOTAL} &gt; {HH1} then goto TooManyYoung; if {HHType} == 'Both' then goto HH4both;</p> <p>integer;NODKRE;</p>	HH3
TOOMANYYOUNG	<p style="text-align: right;">INFO</p> <p>[YOU CAN'T/SHOULDN'T HAVE MORE PEOPLE LESS THAN 5 YEARS OLD AND GREATER THAN OR EQUAL TO 15 YEARS OLD THAN THERE ARE TOTAL HH MEMBERS; PLEASE DOUBLE CHECK THE ANSWERS SO FAR.]</p> <p>backupto HH2A;</p>	HH2A

Question Name	Question Attributes	Next Question
HH3	<p>NUM</p> <p>How many {HHTypeLowCaseENG} household members are 15 years of age or older?&lt;BIG&gt;</p> <p>[Range: 0 - 20]</p> <p>if {HH3} == 0 then goto NoEligible; if {HH3} &gt; {HH2} then goto TooMany2;</p> <p>integer;NODKRE;</p>	HH4
TOOMANY2	<p>INFO</p> <p>[YOU CAN'T/SHOULDN'T HAVE MORE {HHTypeUpCaseENG}S GREATER THAN OR EQUAL TO 15 THAN THERE ARE TOTAL HH MEMBERS GREATER THAN OR EQUAL TO 15. PLEASE DOUBLE CHECK THE ANSWERS SO FAR.]</p> <p>backupto HH3;</p>	HH3
HH4	<p>INFO</p> <p>I now would like to collect information about the {HH3} {HHTypeLowCaseENG}(s) that live in this household who are 15 years of age or older.&lt;P&gt;</p> <p>Let's start listing the {HHTypeLowCaseENG}(s) from oldest to youngest.</p> <p>set {Fill1ENG_0} ='oldest'; set {Fill1LNG_0} ='oldest [IN SECOND LANGUAGE]';</p>	RLOOP
HH4BOTH	<p>INFO</p> <p>I now would like to collect information about only these persons that live in this household who are 15 years of age or older.&lt;P&gt;</p> <p>Let's start listing them from oldest to youngest.</p> <p>set {Fill1ENG_0} ='oldest'; set {Fill1LNG_0} ='oldest [IN SECOND LANGUAGE]'; StoreVariable(HH3);</p>	Rloop
RLOOP	<p>LOOP</p>	HH4a
HH4A	<p>TEXT</p> <p>What is the {Fill1ENG_0} person's first name?</p> <p>NODK;</p>	HH4B

Question Name	Question Attributes	Next Question																																													
HH4B	<p>What is this person's age?&lt;P&gt; [IF RESPONDENT DOESN'T KNOW, PROBE FOR AN ESTIMATE]&lt;BIG&gt;</p> <p>[Range: 15 - 110]</p> <p>if ({HH4b} &gt;=15) &amp;&amp; ({HH4b} &lt;= 17) then goto HH4c;</p> <p>integer;NODKRE;</p>	HH4DCOMP																																													
HH4C	<p>What is the month of this person's date of birth?</p> <table border="1"> <thead> <tr> <th>Seq</th><th>Code</th><th>English</th></tr> </thead> <tbody> <tr><td>1</td><td>1</td><td>01</td></tr> <tr><td>2</td><td>2</td><td>02</td></tr> <tr><td>3</td><td>3</td><td>03</td></tr> <tr><td>4</td><td>4</td><td>04</td></tr> <tr><td>5</td><td>5</td><td>05</td></tr> <tr><td>6</td><td>6</td><td>06</td></tr> <tr><td>7</td><td>7</td><td>07</td></tr> <tr><td>8</td><td>8</td><td>08</td></tr> <tr><td>9</td><td>9</td><td>09</td></tr> <tr><td>10</td><td>10</td><td>10</td></tr> <tr><td>11</td><td>11</td><td>11</td></tr> <tr><td>12</td><td>12</td><td>12</td></tr> <tr><td>13</td><td>-7</td><td>DON'T KNOW</td></tr> <tr><td>14</td><td>-9</td><td>REFUSED</td></tr> </tbody> </table>	Seq	Code	English	1	1	01	2	2	02	3	3	03	4	4	04	5	5	05	6	6	06	7	7	07	8	8	08	9	9	09	10	10	10	11	11	11	12	12	12	13	-7	DON'T KNOW	14	-9	REFUSED	HH4CYEAR
Seq	Code	English																																													
1	1	01																																													
2	2	02																																													
3	3	03																																													
4	4	04																																													
5	5	05																																													
6	6	06																																													
7	7	07																																													
8	8	08																																													
9	9	09																																													
10	10	10																																													
11	11	11																																													
12	12	12																																													
13	-7	DON'T KNOW																																													
14	-9	REFUSED																																													
HH4CYEAR	<p>What is the year of this person's date of birth?</p> <p>[Range: 1910 - 2010]</p> <p>if {HH4c} == -7    {HH4c} == -9    {HH4cYear} == -7    {HH4cYear} == -9 then goto HH4dComp; call ValidateBDay; if {calcyears} &lt; 15 then goto ValidateAge;</p> <p>integer;NODKRE,RngInclude=-7;-9;</p>	HH4DComp																																													
VALIDATEAGE	<p>[AGE CALCULATED FROM BIRTH MONTH AND BIRTH YEAR IS LESS THAN 15. PLEASE DOUBLE CHECK THESE ANSWERS.]</p> <p>backupto HH4C;</p>	HH4C																																													

Question Name	Question Attributes	Next Question															
HH4DCOMP	<p style="text-align: right;">COMP</p> <p>set {MFFillENG_0} = '[RECORD GENDER (FOR VERIFICATION IF NECESSARY)]';</p> <p>if {HHType_0} == 'Both' then set {MFFillENG_0} = 'Is this person male or female?';</p> <p>set {MFFillLNG_0} = '[RECORD GENDER (FOR VERIFICATION IF NECESSARY) IN SECOND LANGUAGE]';</p> <p>if {HHType_0} == 'Both' then set {MFFillLNG_0} = 'Is this person male or female? [IN SECOND LANGUAGE]';</p>	HH4D															
HH4D	<p style="text-align: right;">MALEFEM LIST</p> <p>{MFFillENG_0}</p> <table border="1"> <thead> <tr> <th>Seq</th><th>Code</th><th>English</th></tr> </thead> <tbody> <tr> <td>1</td><td>1</td><td>MALE</td></tr> <tr> <td>2</td><td>2</td><td>FEMALE</td></tr> </tbody> </table> <p>IF {HHType_0}== 'Both' THEN GOTO HH4e;  If {HH4d} == 1 &amp;&amp; {HHType_0} == 'Female' then goto WrongGender;  If {HH4d} == 2 &amp;&amp; {HHType_0} == 'Male' then goto WrongGender;</p>	Seq	Code	English	1	1	MALE	2	2	FEMALE	HH4E						
Seq	Code	English															
1	1	MALE															
2	2	FEMALE															
WRONGGENDER	<p style="text-align: right;">INFO</p> <p>[YOU SHOULD BE ROSTERING ONLY {HHTypeUpCaseENG_0}S; PLEASE DOUBLE CHECK THIS DATA.]</p> <p>backupto HH4D;</p>	HH4D															
HH4E	<p style="text-align: right;">YESNODKREF LIST</p> <p>Does this person currently smoke tobacco, including  &lt;B&gt;&lt;I&gt;[FILL APPROPRIATE COUNTRY EXAMPLES:  cigarettes, cigars, pipes, waterpipe]&lt;/B&gt;&lt;/I&gt;?</p> <table border="1"> <thead> <tr> <th>Seq</th><th>Code</th><th>English</th></tr> </thead> <tbody> <tr> <td>1</td><td>1</td><td>YES</td></tr> <tr> <td>2</td><td>2</td><td>NO</td></tr> <tr> <td>3</td><td>-7</td><td>DON'T KNOW</td></tr> <tr> <td>4</td><td>-9</td><td>REFUSED</td></tr> </tbody> </table> <p>set {Fill1ENG_0} = 'next oldest';  set {Fill1LNG_0} = 'next oldest [IN SECOND LANGUAGE]';</p>	Seq	Code	English	1	1	YES	2	2	NO	3	-7	DON'T KNOW	4	-9	REFUSED	LoopEnd
Seq	Code	English															
1	1	YES															
2	2	NO															
3	-7	DON'T KNOW															
4	-9	REFUSED															
LOOPEND	<p style="text-align: right;">LEND</p>	EditRosterIntro															

Question Name	Question Attributes	Next Question
EDITROSTERINTRO	<p style="text-align: right;">INFO</p> <p>[IF YOU NEED TO REVIEW THE ROSTER, SELECT THE ROSTER ICON.&lt;P&gt;</p> <p>GO BACK IF YOU NEED TO MAKE CHANGES. &lt;P&gt;</p> <p>GO TO NEXT SCREEN TO SELECT THE RESPONDENT.]</p>	SELECT
SELECT	<p style="text-align: right;">COMP</p> <p>Call GatsSelAlgo;</p> <p>Call GATSSaveUserInfo;</p>	HH5
HH5	<p style="text-align: right;">INFO</p> <p>[NAME OF THE SELECTED ELIGIBLE PERSON IS:&lt;P&gt;</p> <p>{SelName}&lt;P&gt;</p> <p>ASK IF THE SELECTED RESPONDENT IS AVAILABLE AND IF SO, PROCEED TO THE INDIVIDUAL QUESTIONNAIRE. &lt;P&gt;</p> <p>IF THE SELECTED RESPONDENT IS NOT AVAILABLE, MAKE AN APPOINTMENT AND RECORD IT AS A COMMENT IN THE VISIT RECORD.]</p> <p>set {EventCode} = 200;</p> <p>set {EventComment} = 'Screener Complete';</p> <p>NoBackup,NoBreakOff;</p>	CODEEVENTS
NOELIGIBLE	<p style="text-align: right;">INFO</p> <p>[THERE ARE NO ELIGIBLE HOUSEHOLD MEMBERS.&lt;P&gt;</p> <p>THANK THE RESPONDENT FOR HIS/HER TIME.&lt;P&gt;</p> <p>YOU WILL NOT BE PERMITTED TO BACK UP FROM THE NEXT SCREEN.]</p>	NOELIGIBLE2
NOELIGIBLE2	<p style="text-align: right;">INFO</p> <p>[THIS WILL BE RECORDED IN THE VISIT RECORD AS A CODE 201.]</p> <p>set {EventCode} = 201;</p> <p>set {EventComment} = 'Screener Complete No Eligibles';</p> <p>NoBackup,NoBreakOff;</p>	CODEEVENTS
CODEEVENTS	<p style="text-align: right;">COMP</p> <p>call Code_Event;</p> <p>if {EventCode} == 200 then call Activate_Case;</p>	HHENDSURVEY



Question Name	Question Attributes	Next Question
HHENDSURVEY	[END OF INTERVIEW] NoBackup,NoBreakOff;	END
END	[TAP EXIT BUTTON TO END] NoBackup,NoBreakOff;	Null

As described previously, these specifications are stored in SQLite tables (Questions *Exhibit 3-11*, Texts *Exhibit 3-12*, Answers *Exhibit 3-13*). The screen types used to implement this questionnaire are Comp, Info, Num, List, Text, Loop, and Loop End. The visible screens generated by these specifications and a sample set of specific answers provided are shown as screenshots following *Exhibit 3-13*. (Comp screens do not display any information to the Field Interviewer, and depending upon how questions are answered, many paths are possible through the questionnaire and this is only one possible path.) Specific lines of the Questions table (*Exhibit 3-11*) reinforce features introduced before and illustrate several new features. Specific details to note are:

- The Q sequence numbers are not sequential or in any specific order, but they are unique. They serve only to order the display of the rows in the Questions table when developing questionnaires. The actual order of the questions asked by the FI is determined by beginning with the Start screen and then following the logic dictated by each question.
- The Instance Level is shown on the top line of the screen after the QID in square brackets. It is 0 when not in a loop and >0 when inside a loop. It can be seen to be changing from 0 in the loop over members of the household. (See screenshots HH4a–HH4e below.)
- The QID names are ≤20 characters and are unique within a given survey.
- The screenshot TooMany illustrates the result of a "wrong" answer to the previous question where the respondent said there were more 15-year-old people than there were household members. This error is trapped by the program logic in the QLogic block for QID HH2; for this example, we assume that the FI corrected the error when sent back to screen HH2 and then moved forward with the interview.
- A start screen should be a Comp-type screen, so it is not displayed; note that in the Texts table the text for the **Start** question is the questionnaire version number. The version number is stored there by the designer program.
- QID Fills shows how the GSS set statement is used to create variables used as fills. For example, set {COUNTRYNAME}= 'Country' creates a variable that can be used anywhere in the questionnaire and is used as a fill in QID Intro1.
- QID Intro1 shows the use of a variable as a fill; when the Intro screen is displayed to the FI, the {Agency} variable is filled by its current value as shown in the Intro1 screenshot (See *Exhibit 3-12* for QID Intro1 text specification).
- QID HH1 shows how an if-then syntax is used to manage flow control in a questionnaire. In this screen, if there are no eligible people (i.e., the answer to HH1 is 0), then you would skip to QID NoEligible. HH1 also illustrates the use of a range, in this case a range of 0 to 50 (RangeLo=0

and RangeHi=50). Also, the Special Instructions field for this screen specifies that only integer values will be accepted as answers to HH1.

- QID HH2 illustrates the use of three multiple logic statements.
- QID Rloop illustrates the use of the Loop-type screen to start a loop over the number of people in the household. The loop should start at 1 (field LoopStart) and go to the value of HH3 (field LoopEnd). For the Loop-type screen, the QNext field tells the GSS what screen or QID to use to start the loop.
- QID HH4b illustrates the use of numeric comparisons in the QLogic field. When performing logical comparisons in the GSS, you can use character or numeric comparisons.
- QID LoopEnd illustrates the use of the LoopEnd-type of screen. This screen type demarcates the end of a loop, in this case the loop over household members. It has two required fields: LoopEnd and LoopGoTo. The LoopEnd field is a number or variable that specifies the end limit value of the loop. The LoopGoTo is the QID of the screen to go to when the loop counter is less than or equal to this limit (LoopEnd value), and the QNext field is the QID of the screen to go to when the loop counter is greater than this limit (LoopEnd value), that is, when the loop is complete.
- QID Select is an example of calling user-supplied subroutines. The Call statements call subroutines that have been added to GSS for GATS. The GatsSelAlgo routine is the sample selection routine and the GATSSaveUserInfo is a special purpose routine that saves intermediate data (e.g., the random number from the sampling routine). These routines are coded in Java and added to the GSS build.
- QID CodeEvents is a Comp-type screen that shows an example of how you can auto code events depending on the questionnaire data. Here, if certain data conditions are met, the case is event coded, and in addition, in other conditions both the HQ case and the IQ case are event coded. The Code\_Event and Code\_EventIQ are GATS subroutines that write event codes to the relevant GSS data tables (Dwelling Unit Event Table [DUEVT]).

**Exhibit 3-11. Screen Information: Questions Table**

Q	Sequence	Qid	QNext	QLogic	QType	Format	Qanswer	Range Lo	Range Hi	Loop Start	Loop End	Loop GoTo	Roster #	Roster Col
1	START		INTRO		COMP		None	0	0				0	0
21	INTRO		INTRO1		INFO	PASSIVE_GPS_ CAPTURE;	None	0	0				0	0
21.05	INTRO_ PASSIVE_GPS		END		TEXT		None	0	0				0	0
31	INTRO1		HH1	set {HHTYPE} = '{Parm2}'; set {HHTYPELowCaseENG} = 'both'; if {HHTYPE} == 'Male' then set {HHTYPELowCaseENG} = 'male'; if {HHTYPE} == 'Female' then set {HHTYPELowCaseENG} = 'female'; set {HHTYPEUpCaseENG} = 'BOTH'; if {HHTYPE} == 'Male' then set {HHTYPEUpCaseENG} = 'MALE'; if {HHTYPE} == 'Female' then set {HHTYPEUpCaseENG} = 'FEMALE';	INFO		None	0	0				0	0
41	HH1		HH2	if {HH1} == 0 then goto NoEligible;	NUM	integer;NODKRE;	None	0	50				0	0
51	HH2		HH2A	if {HH2} == 0 then goto NoEligible; if {HH2} > {HH1} then goto TooMany; if {HHTYPE} == 'Both' then set {HH3} = {HH2}; if {HH2} == {HH1} then goto HH3; if {HHTYPE} == 'Both' then goto HH4both;	NUM	integer;NODKRE;	None	0	20				0	0
61	TOOMANY		HH2	backupto HH2;	INFO		None	0	0				0	0
65	HH2A		HH3	SET {HHTOTAL} = {HH2} + {HH2A}; if {HHTOTAL} > {HH1} then goto TooManyYoung;	NUM	integer;NODKRE;	None	0	20				0	0
66	TOOMANYYOUNG		HH2A	backupto HH2A;	INFO		None	0	0				0	0
71	HH3		HH4	if {HH3} == 0 then goto NoEligible; if {HH3} > {HH2} then goto TooMany2;	NUM	integer;NODKRE;	None	0	20				0	0
81	TOOMANY2		HH3	backupto HH3;	INFO		None	0	0				0	0

Q	Sequence Qid	QNext	QLogic	QType	Format	Qanswer	Range Lo	Range Hi	Range Start	Loop End	Loop GoTo	Roster #	Roster Col
91	HH4	RLOOP	set {Fill1ENG_0} = 'oldest'; set {Fill1LING_0} = 'oldest' [IN SECOND LANGUAGE];	INFO		None	0	0			0	0	0
101	HH4BOTH	Rloop	set {Fill1_0} = 'oldest';	INFO		None	0	0			0	0	0
111	RLOOP	HH4a		LOOP		None	0	0	1	{HH3_0}	0	0	0
121	HH4A	HH4B		TEXT	NODK;	None	0	0			1	0	0
131	HH4B	HH4DCOMP	if ({HH4b} >= 15) && ({HH4b} <= 17) then goto HH4c;	NUM	integer;NODKRE;	None	15	110			1	1	1
141	HH4C	HH4CYEAR		LIST		MONTHS	0	0			1	2	
151	HH4CYEAR	HH4DComp	if {HH4c} == -7    {HH4c} == -9    {HH4cYear} == -7    {HH4cYear} == -9 then goto HH4dComp; call ValidateBDay; if {calcyers} < 15 then goto ValidateAge;	NUM	integer;NODKRE, RngInclude=-7,-9;	DKREF	1905	2005			1	3	
161	VALIDATEAGE	HH4C	backupto HH4C;	INFO		None	0	0			0	0	0
165	HH4DCOMP	HH4D	set {MFFillENG_0} = 'RECORD GENDER (FOR VERIFICATION IF NECESSARY)';  if {HHType_0} == 'Both' then set {MFFillENG_0} = 'Is this person male or female?'; set {MFFillLING_0} = 'RECORD GENDER (FOR VERIFICATION IF NECESSARY) IN SECOND LANGUAGE';  if {HHType_0} == 'Both' then set {MFFillLING_0} = 'Is this person male or female?' [IN SECOND LANGUAGE];	Comp		None	0	0			0	0	0

Q	Sequence	Qid	QNext	Qlogic	QType	Format	Qanswer	Range Lo	Range Hi	Range Start	Loop End	Loop GoTo	Roster #	Roster Col
171	HH4D	HH4E	HH4E	IF {HHTYPE_0 }== 'Both' THEN GOTO HH4E; IF {HH4d} == 1 && {HHTYPE_0} == 'Female' then goto WrongGender; IF {HH4d} == 2 && {HHTYPE_0} == 'Male' then goto WrongGender; backupto HH4D;	LIST		MALEFEM	0	0			1	4	
181	WRONGGENDER	HH4D			INFO		None	0	0			0	0	
191	HH4E	LoopEnd		set {Fill1ENG_0} = 'next oldest';  set {Fill1LNG_0} ='next oldest [IN SECOND LANGUAGE]';	LIST		YESNODKREF	0	0			1	5	
201	LOOPEND	EditRosterIntro			LEND		None	0	0	1	{HH3_ HH4a 0}	0	0	
211	EDITROSTERINTR O	SELECT			INFO		YESNO	0	0			0	0	
231	SELECT	HH5		Call GatsSelAlgo;  Call GATSSaveUserInfo;	Comp		None	0	0			0	0	
241	HH5	CODEEVENTS		set {EventCode} = 200;  set {EventComment} = 'Screener Complete';	INFO	NoBackup, NoBreakOff;	None	0	0			0	0	
251	NOELIGIBLE	NOELIGIBLE2			INFO		None	0	0			0	0	
252	NOELIGIBLE2	CODEEVENTS		set {EventCode} = 201;  set {EventComment} = 'Screener Complete No Eligibles';	INFO	NoBackup,NoBrea kOff;	None	0	0			0	0	
261	CODEEVENTS	END		call Code_Event;  if {EventCode} == 200 then call Activate_Case;	Comp		None	0	0			0	0	
271	END	Null			END		None	0	0			0	0	

**Exhibit 3-12. GSS Implementation of the GATS Household Questionnaire: Texts Table**

QID	Language	QText
CodeEvents	ENGLISH	
EditRosterIntro	ENGLISH	[IF YOU NEED TO REVIEW THE ROSTER, SELECT "ROSTER" FROM THE TOOLS MENU. SWIPE BACK IF YOU NEED TO MAKE CHANGES. SWIPE FORWARD TO SELECT THE RESPONDENT.]
END	ENGLISH	DO YOU REALLY WANT TO END THE INTERVIEW? IF SO, TAP Exit TO END, OTHERWISE SWIPE BACK TO GO BACK.
Fills	ENGLISH	
HH1	ENGLISH	First, I'd like to ask you a few questions about your household. In total, how many persons live in this household? [INCLUDE ANYONE WHO CONSIDERS THIS HOUSEHOLD THEIR USUAL PLACE OF RESIDENCE]
HH2	ENGLISH	How many of these household members are 15 years of age or older?
HH3	ENGLISH	How many {HHTYPE} household members are 15 years of age or older?
HH4	ENGLISH	I now would like to collect information about the {HH3} {HHTYPE}(s) that live in this household who are 15 years of age or older. Let's start listing the {HHTYPE}(s) from oldest to youngest.
HH4a	ENGLISH	What is the {Fill1_0} person's first name?
HH4b	ENGLISH	What is this person's age?
HH4both	ENGLISH	[IF RESPONDENT DOESN'T KNOW, PROBE FOR AN ESTIMATE] I now would like to collect information about only these persons that live in this household who are 15 years of age or older. Let's start listing them from oldest to youngest.
HH4c	ENGLISH	What is the month of this person's date of birth?
HH4cYear	ENGLISH	What is the year of this person's date of birth?
HH4d	ENGLISH	{MFFill_0}
HH4dComp	ENGLISH	
HH4e	ENGLISH	Does this person currently smoke tobacco, including {CountryExamples_0}?

QID	Language	QText
HH5	ENGLISH	[NAME OF THE SELECTED ELIGIBLE PERSON IS: {SelName} ASK IF THE SELECTED RESPONDENT IS AVAILABLE AND IF SO, PROCEED TO THE INDIVIDUAL QUESTIONNAIRE. IF THE SELECTED RESPONDENT IS NOT AVAILABLE, MAKE AN APPOINTMENT AND RECORD IT AS A COMMENT ON RECORD OF VISITS.]
Intro	ENGLISH	[THE HOUSEHOLD SCREENING RESPONDENT MUST BE 18 YEARS OF AGE OR OLDER AND YOU MUST BE CONFIDENT THAT THIS PERSON CAN PROVIDE ACCURATE INFORMATION ABOUT ALL MEMBERS OF THE HOUSEHOLD. IF NEEDED, VERIFY THE AGE OF THE HOUSEHOLD SCREENING RESPONDENT TO MAKE SURE HE/SHE IS 18 YEARS OF AGE OR OLDER.]
Intro1	ENGLISH	An important survey of adult tobacco use behavior is being conducted by the {Agency} throughout {CountryName} and your household has been selected to participate. All houses selected were chosen from a scientific sample and it is very important to the success of this project that each participates in the survey. All information gathered will be kept strictly confidential. I have a few questions to find out who in your household is eligible to participate.
NoEligible	ENGLISH	[THERE ARE NO ELIGIBLE HOUSEHOLD MEMBERS. THANK THE RESPONDENT FOR HIS/HER TIME. THIS WILL BE RECORDED IN THE RECORD OF VISITS AS A CODE 201.]
Rloop	ENGLISH	Ver.: 2019.04.17.043 [YOU CAN'T/SHOULDN'T HAVE MORE PEOPLE >= 15 YEARS OLD THAN THERE ARE TOTAL HH MEMBERS; PLEASE DOUBLE CHECK THE ANSWERS SO FAR.] [YOU CAN'T/SHOULDN'T HAVE MORE {HType}s >=15 THAN THERE ARE TOTAL HH MEMBERS >=15. PLEASE DOUBLE CHECK THE ANSWERS SO FAR.] [YOU SHOULD NOT BE ROSTERING PEOPLE BELOW THE AGE OF 15. PLEASE DOUBLE CHECK THE DATA ENTERED FOR THIS PERSON.] [AGE CALCULATED FROM BIRTH MONTH AND BIRTH YEAR IS LESS THAN 15. PLEASE DOUBLE CHECK THESE ANSWERS.]
Select	ENGLISH	
Start	ENGLISH	
TooMany	ENGLISH	
TooMany2	ENGLISH	
TooYoung	ENGLISH	
ValidateAge	ENGLISH	
WrongGender	ENGLISH	[YOU SHOULD BE ROSTERING ONLY {HType_0}s; PLEASE DOUBLE CHECK THIS DATA.]

*Exhibit 3-13* lists the answer sets that are specified in the Answers table. An answer set is a list of option choices for a specific List or ATA-type question. Each list or ATA-type screen must have an answer set, but you can use the same answer set for many screens. An answer set must specify the sequence in which to present the choices, the codes to store in the database if a choice is selected, and the text to display for each choice. *Exhibit 3-12* shows the answer sets required for this questionnaire. The AID column is the ID of the answer set or its name; the ASequence column tells GSS what order to display the possible answer choices; the ACode column is the code that is stored in the database (in the responses table) when an answer choice is selected; the AText column is the text displayed; and the Language column identifies the language of this answer set. An answer set has a row for each choice presented to the FI. There are four answer sets defined in *Exhibit 3-12* with AIDs: MaleFem, Months, YesNo, and YesNODKREF. These are arbitrary names for the answer sets, but they must be unique within a questionnaire specification. Answer sets can be used on more than one question or none at all. (If you use sets from prior questionnaire building, you can carry them over even if you do not use them.) The answer sets are used in the following screens:

- HH4d uses the MaleFem answer set.
- EditRosterIntro uses the YESNO answer set.
- HH4e uses the YesNoDKREF answer set.
- HH4c uses the Months answer set.

The link between the Answers table (AID column) and the Questions table is the QAnswer variable or column in the Questions Table. (See *Exhibit 3-11* to see where the answer sets are specified for these questions.)

The other screens do not use answer sets, because they are not List, Numeric or ATA-type screens and they are specified as none in the QAnswer column of the Questions table. The AWav column is not used in GATS. Since this example uses only English, the Answers table shows only one language. If multiple languages were used, then each language would have a set of rows with the same data items but with the text items translated into the desired language. For example, if the MaleFem answer set was to be in French, we would also see the following rows in the Answers table:

900	MALEFEM	French	1	1	Homme
901	MALEFEM	French	2	2	Femme



**Exhibit 3-13. GSS Implementation of the GATS HQ: Answers Table**

Answers					
UID	AID	Language	ASequence	ACode	AText
338	YESNO	ENGLISH	1	1	YES
339	YESNO	ENGLISH	2	2	NO
412	YESNODKREF	ENGLISH	1	1	YES
413	YESNODKREF	ENGLISH	2	2	NO
414	YESNODKREF	ENGLISH	3	-7	DON'T KNOW
415	YESNODKREF	ENGLISH	4	-9	REFUSED
464	MONTHS	ENGLISH	1	1	01
465	MONTHS	ENGLISH	2	2	02
466	MONTHS	ENGLISH	3	3	03
467	MONTHS	ENGLISH	4	4	04
468	MONTHS	ENGLISH	5	5	05
469	MONTHS	ENGLISH	6	6	06
470	MONTHS	ENGLISH	7	7	07
471	MONTHS	ENGLISH	8	8	08
472	MONTHS	ENGLISH	9	9	09
473	MONTHS	ENGLISH	10	10	10
474	MONTHS	ENGLISH	11	11	11
475	MONTHS	ENGLISH	12	12	12
476	MONTHS	ENGLISH	13	-7	DON'T KNOW
477	MONTHS	ENGLISH	14	-9	REFUSED
478	MALEFEM	ENGLISH	1	1	MALE
479	MALEFEM	ENGLISH	2	2	FEMALE

The GSS specifications provided above leads to the following set of screens for this example, given the specific answers shown on the screens. This example assumes that the "Type" variable that specifies the household sampling level is "Male." Other possible screens are skipped due to the routing path as dictated by the answers. The questions have been answered to show a household with two males over 15. They are Arnold, a 19-year-old, and Bill, a 33-year-old nonsmoker (see screens HH4A–HH4E, levels 1 and 2). Arnold has been randomly selected to continue on with the GATS *Individual Questionnaire* (see Screen HH5).

Intro

INTRO [0] 🗨️

[THE HOUSEHOLD SCREENING RESPONDENT SHOULD BE 18 YEARS OF AGE OR OLDER AND YOU MUST BE CONFIDENT THAT THIS PERSON CAN PROVIDE ACCURATE INFORMATION ABOUT ALL MEMBERS OF THE HOUSEHOLD. IF NEEDED, VERIFY THE AGE OF THE HOUSEHOLD SCREENING RESPONDENT TO MAKE SURE HE/SHE IS 18 YEARS OF AGE OR OLDER.

THE HOUSEHOLD SCREENING RESPONDENT CAN BE LESS THAN 18 YEARS OLD, ONLY IF NO HOUSEHOLD MEMBERS ARE 18 YEARS OF AGE OR OLDER.]

Navigation icons: back, stop, fast forward, next.

Intro1

INTRO1 [0] 🗨️

An important survey of adult tobacco use behavior is being conducted by the **[FILL COUNTRY SPONSORING AGENCY]** throughout **[FILL COUNTRY]** and your household has been selected to participate. All houses selected were chosen from a scientific sample and it is very important to the success of this project that each participates in the survey. All information gathered will be kept strictly confidential. I have a few questions to find out who in your household is eligible to participate.

Navigation icons: back, stop, fast forward, next.

HH1

HH1 [0] 🗨️

First, I'd like to ask you a few questions about your household. In total, how many persons live in this household?

[INCLUDE ANYONE WHO CONSIDERS THIS HOUSEHOLD THEIR USUAL PLACE OF RESIDENCE]

Navigation icons: back, stop, fast forward, next.

Keypad: 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, Done, backspace.

HH2

HH2 [0] 🗨️

How many of these household members are 15 years of age or older?

Navigation icons: back, stop, fast forward, next.

Keypad: 1, 2, 3, 4, 5, 6, 7, 8, 9, 0, Done, backspace.

HH4both

Survey

HH4both [0]

I now would like to collect information about only these persons that live in this household who are 15 years of age or older.

Let's start listing them from oldest to youngest.

HH4a

Survey

HH4a [1]

What is the oldest person's first name?

HH4B

Survey

HH4B [1]

What is this person's age?

[IF RESPONDENT DOESN'T KNOW, PROBE FOR AN ESTIMATE]

HH4D

Survey

HH4D [1]

[RECORD GENDER (FOR VERIFICATION IF NECESSARY)]

☐ MALE

☐ FEMALE

HH4E

Survey

HH4E [1]

Does this person currently smoke tobacco, including **[FILL APPROPRIATE COUNTRY EXAMPLES: cigarettes, cigars, pipes, waterpipe]**?

☐ YES  
☐ NO  
☐ DON'T KNOW  
☐ REFUSED

HH4A

Survey

HH4a [2]

What is the next oldest person's first name?

1 2 3 4 5 6 7 8 9 0 Del  
 q w e r t y u i o p ⌫  
 a s d f g h j k l Done  
 ↑ z x c v b n m , . ? ↑  
 Ctrl !#1 English (UK) ⌵ ▶

HH4B

Survey

HH4B [2]

What is this person's age?

[IF RESPONDENT DOESN'T KNOW, PROBE FOR AN ESTIMATE]

1 2 3 ⌫  
 4 5 6 Done  
 7 8 9 -  
 0 .

HH4D

Survey

HH4D [2]

Is this person male or female?

☐ MALE  
☐ FEMALE

HH4e

Survey

HH4e [2]

Does this person currently smoke tobacco, including **[FILL APPROPRIATE COUNTRY EXAMPLES: cigarettes, cigars, pipes, waterpipe]**?

☐ YES  
☐ NO  
☐ DON'T KNOW  
☐ REFUSED

EditRosterIntro

Survey

EditRosterIntro [0]

[IF YOU NEED TO REVIEW THE ROSTER, SELECT "ROSTER" FROM THE TOOLS MENU.]

GO BACK IF YOU NEED TO MAKE CHANGES.

GO TO NEXT SCREEN TO SELECT THE RESPONDENT.]

HH5

Survey

HH5 [0]

[NAME OF THE SELECTED ELIGIBLE PERSON IS:

Arnold

ASK IF THE SELECTED RESPONDENT IS AVAILABLE AND IF SO, PROCEED TO THE INDIVIDUAL QUESTIONNAIRE.

IF THE SELECTED RESPONDENT IS NOT AVAILABLE, MAKE AN APPOINTMENT AND RECORD IT AS A COMMENT ON RECORD OF CALLS.]

HHEndSurvey

Survey

HHEndSurvey [0]

[END OF INTERVIEW]



## 4. GSS Tool Kit

---

This chapter explains the functionality of a set of software products that support the development of forms in GSS. The GSS Tool Kit is a suite of Windows OS-based PC programs designed to facilitate developing GSS forms and managing GSS databases. The Tool Kit starts with a main control program that serves as the main menu or pathway to the GSS suite of Windows PC-based tools.

The GSS Tool Kit organizes Windows PC-based GSS modules that support questionnaire development and data management into a single user interface to make access and use of these programs easier and simpler. The screenshot below shows the initial or startup view.

*Start Screen of Main Menu for GSS Developer Tools*



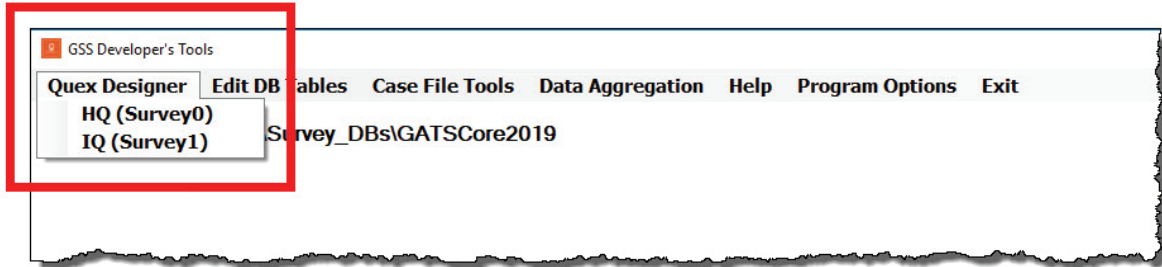
The systems were developed on standard PC platforms running Windows. The Tool Kit is designed for PC laptops with screen resolution at or below 1024 x 768. The .Net framework modules 2.0 or later is used and needs to be on any target platform.

The menu system allows the GATS developer to access the GSS suite of tools from one location. The developer can specify the locations of key folders one time and save that definition for the next use of the tool. Each menu and its functions are described in the following sections.

## 4.1 Questionnaire Designer (Quex Designer)

This module initiates the Designer program for a specific survey form—either HQ (Survey 0) or IQ (Survey 1)—as specified by the setup parameters of the main menu.

*Questionnaire Designer Menu*



The Questionnaire Designer (Quex Designer) is a development tool that helps GATS developers create, edit, and maintain GATS survey forms. It allows GATS developers to fill the metadata tables required by GATS forms from an easy-to-use interface. The Designer does not alter the underlying table structure or metadata. Users can still modify GSS tables directly from third-party SQLite managers and/or utilities.

The Questionnaire Designer supports the following:

- Modifications of question and answer texts, types of questions, skip patterns, and logical checks.
- Validation of user inputs where feasible.
- Limited inputs of certain fields when they are not supported by a screen type; for example, ranges cannot be entered for list type questions.
- Selection of a second language from  $n$  languages to display and work in at one time. English will always be Language 0, and you can choose among the other languages defined in the databases for display as the second language.
- Automated version control for questionnaire updates. The developer receives a prompt when the Questionnaire Designer saves a questionnaire program to update the program's version number. (The questionnaire program version number is stored in the text field for the START question). The format will be Ver.: YYYY MM DD nnn, where YYYY is current year, MM is current month, DD is current day, and nnn is a sequential number that starts at 001 and is incremented by one with every save. A sample version number is as follows: Ver.: 2019 01 30 012.
- A preview capability that shows a reasonable approximation of the screen display for a given question. This feature provides only an approximation of the "true" screen display but is helpful in getting a first approximation of how a screen appears.



## Designer Main Screen

The Designer Main Screen has seven sections:

**Seq # and Quest ID** provides the list of screens that make up a given survey. It is the leftmost area of the screen and shows sequence # and a QID name. Users can place the cursor on this list and use the keyboard navigation keys (up, down, page up, page down) to move about the list of QIDs.

**Question Properties** allows the designer to selection the general properties of a question. Each item in this section is explained in the table below.

Item	Description
Sequence # or Seq #:	A number (it can contain decimals) to control display or sort order of questions, for example, 10.2 or 1235. This number affects only the order in which questions are listed in the display grid; it does not affect the logical order of question execution.
Question ID or QID:	The name or ID of a given question. The name must be unique within a survey and $\geq 1$ character and $\leq 20$ characters.
Roster #:	The specific roster (can be 1 through 4). The roster is an internal grid for data storage, where you want a copy of the answer to the question stored.
Roster column:	The column of the roster where you want a copy of the answer to the question stored, the row is determined by the nesting or loop level.
Question Type:	This box allows you to select a type of question from the dropdown list of available question types (e.g., LIST or TEXT or NUM).
Next Question:	Name of the question ID you want to go to next by default (i.e., if no program logic changes the logical flow; must be unique within a survey $\geq 1$ and $\leq 20$ characters).
Question Help:	Help text for the question that Field Interviewers can display during the survey when they tap the QxQ menu button in GSS. At this time, it can be in only one language. If there is no QxQ text, then the QxQ option in the GSS questionnaire screen is grayed out.

Item	Description
Low Range:	Starting value of acceptable range for a numeric-type question (e.g., 100.5).
High Range:	Ending value of acceptable range for a numeric-type question, for example, 200.8; for example, if the Low Range is 100.5 and the High Range is 200.8, then the numeric question will only accept numbers in the range $100.5 \leq x \leq 200.8$ .
Loop Start:	Starting variable of loop counter.
Loop End:	End value of loop.
Loop Goto:	Target QID to go to when loop is not yet finished (i.e., loop current counter < loop end).

**Question Text** contains text for questions that display text. Users can work with up to two languages at a time. Language 0 is always English and Language 1 can be any one of the languages a user has provided in their Texts tables.

**Answer Texts** provides the answer set lists if the question type is List or ATA for Language 0 and Language 1.

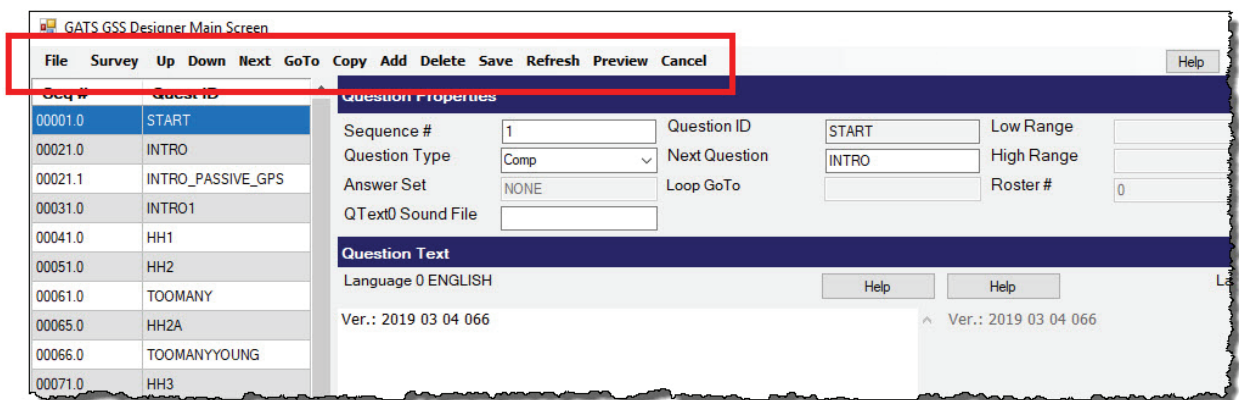
**Special Instructions** allows users to code special format instructions for a question.

**Skip or Compute Logic** allows users to enter their QLogic code. (See *Appendix A* for a discussion of the QLogic syntax and options.)

**QxQ Language**—Question by question language.

Some sections of the Questionnaire Designer have section-specific help that can be invoked.

#### Menu Items

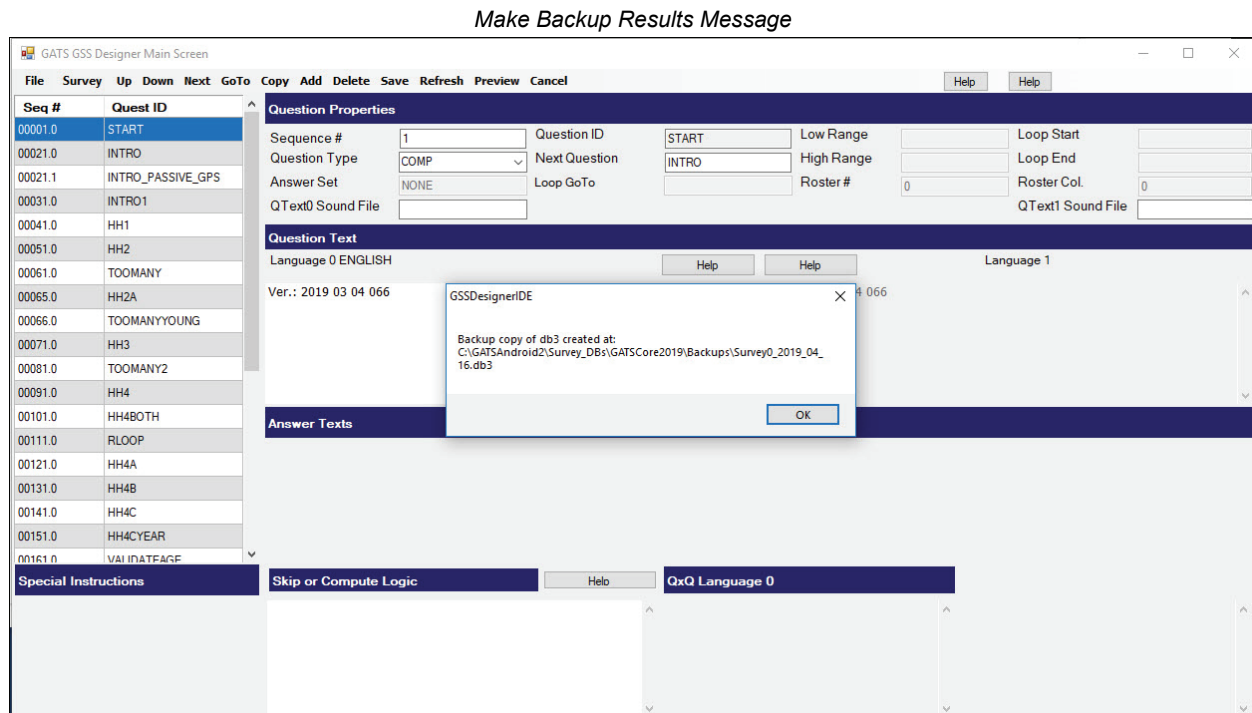


The menu across the top of the Questionnaire Designer screen has multiple actions available: File, Survey, Up, Down, Next, GoTo, Copy, Add, Delete, Save, Refresh, Preview, and Cancel. These actions are described in this section.

Under File in the ribbon menu at the top of the screen, the options include:

- Make Backup, and
- Exit.

The Make Backup option creates a backup of the current database. The naming convention and storage location of the backup are shown in the information box in the following screenshot.



The Survey menu sub-options allow users to select subtasks that affect the entire survey. These are as follows:

- **Resequence:** Causes the sequence numbers (Seq #) in the Questions table to be resequenced using the starting value and increment values that are in properties setup.
- **Properties:** Edit and save the properties data.

### Designer Survey Properties Screen

GSS Survey Properties

Encryption Key

Resequene Start #

Resequene Increment

Survey0 Data Base Path

Survey Data Base Version

SDF Path

**Language Parameters read from INI table** Note: Language parameters are read only.

	Lang. Name	FontName	Font Size
►	ENGLISH	Tahoma	09
*			

Close Form

- **Update Version #:** Update the questionnaire version number stored in the text for the Start QID. The version number will be updated with the current day's date and incremented by one from the previous version. For example, if the current version number is Ver.: 2018 07 06 010 and today is July 6, 2018, then the next or updated version # would be Ver.: 2018 07 06 011.

The other main menu items are:

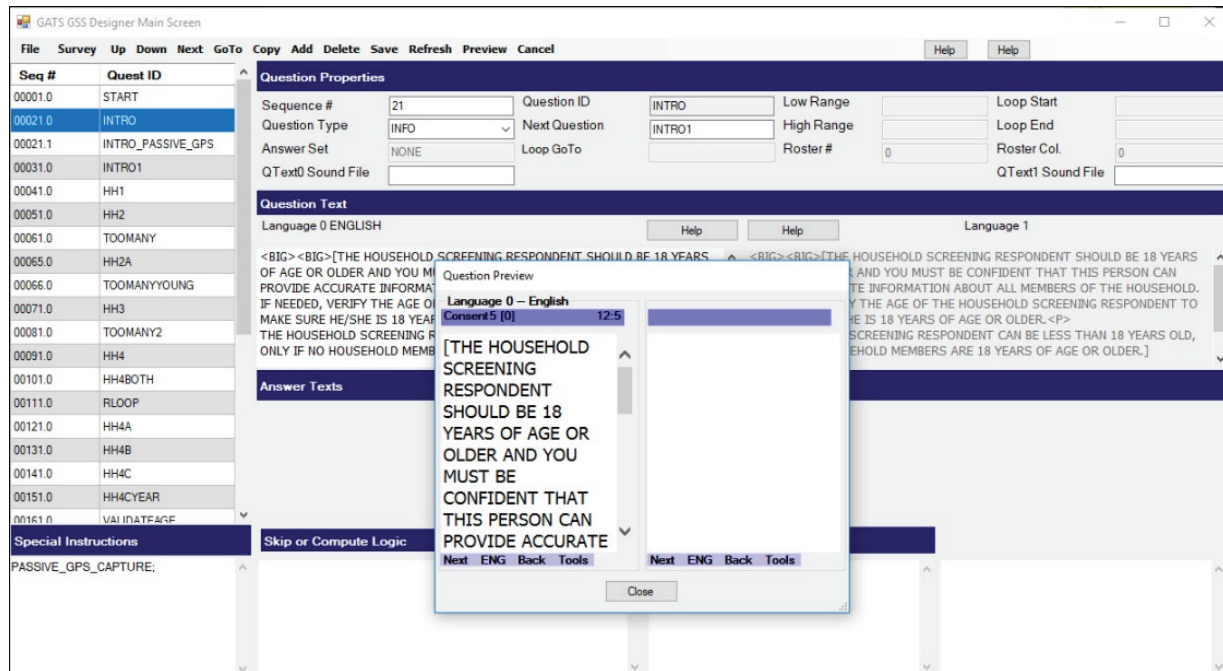
- **Up/Down** allows you to move up or down in the list of QIDs (you can also use the up and down arrows or click on a given QID).
- **Next** goes to the QID specified in the GSS Next Question block.
- **GoTo** prompts you for a specific QID, as shown in the next screenshot.

*Main Designer Screen with GoTo option*

The screenshot displays the 'GATS GSS Designer Main Screen'. On the left is a list of question IDs (Seq #) and their corresponding question IDs (Quest ID). The main area is divided into sections: 'Question Properties' (with fields for Sequence #, Question Type, Answer Set, QText0 Sound File, Question ID, Next Question, Loop GoTo, Low Range, High Range, Roster #, Loop Start, Loop End, Roster Col., and QText1 Sound File), 'Question Text' (with Language 0 ENGLISH and Language 1 sections), 'Answer Texts', 'Special Instructions', 'Skip or Compute Logic', and 'QxQ Language 0'. A small dialog box titled 'Question ID of target' is open in the center, prompting the user to 'Enter the Question ID' with 'Start' entered in the input field.

- **Copy/Add/Delete** allows you to:
  - Add a new QID screen with a new QID,
  - Copy an existing QID screen and rename it, or
  - Delete an existing QID screen.
- **Save** forces an immediate save of the current screen that is displayed. Normally a screen is saved when you move from it to another QID screen.
- **Refresh** causes the current QID screen to be refreshed by reading in the stored data for it.
- **Cancel** ignores the current changes to a QID screen and brings back the QID screen contents as of the last Save. Cancel allows you to cancel any changes made to a screen before you move. It retrieves the saved data from the last time the screen was saved. This allows you to cancel changes you may have made in error or no longer want before moving from the screen, thereby causing the screen to be saved.
- **Preview** displays a preview screen that shows an approximation of what the screen might look like given the current parameters for the QID. A sample preview of a Num type screen is shown in the next screenshot with both English and Mandarin text. Note that hypertext markup language (HTML) codes are not seen in the preview screen.

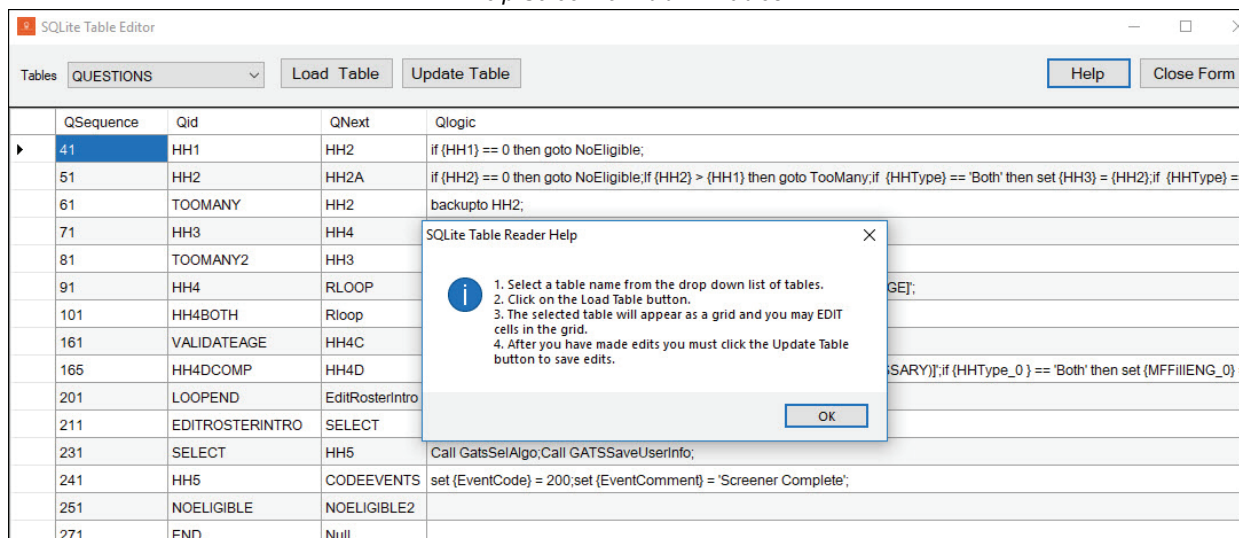
## Preview Screen



## 4.2 Edit DB Tables

Edit DB Tables allows developers to review and edit internal SQLite tables in the Surveys 0 and 1 (Tables: Answers, INI, Questions, QxQ and Texts) databases. The following screenshot shows the Questions table open for Survey 0 and the help file open as well. The user can edit the tables and add new rows. The Help screen in the following screenshot explains how to operate the edit options.

### Help Screen for Edit DB Tables



### 4.3 Case File Tools

CMS Grid Designer is used to control the fields that display in the CMS case grid.

The CMS Grid Designer allows the developer to manage the fields that are displayed in the CMS grid and on the Case Details screen. The developer can drag and drop from the list of Case File Variables to the Case Item Layout. The developer can include extra punctuation or space between fields that will display. For example, to display City, State, ZIP Code, the developer might add [DATA10], [DATA11] [DATA12] to a row of the display.

Under Case Info, the user can manage the fields that are displayed in Case Details. To add a field, the developer can drag and drop a field onto the Case Info grid. To change the order of display, the developer can update the numbers in the Row Order column. To change the labels, the developer can update the text in the "Column Label" column for multiple languages.

*CMS Grid Designer: First Screen*

The screenshot displays the 'CMS Grid Designer' application window. At the top, there is a 'File' menu and a 'Select Language' dropdown set to 'ENGLISH'. The main area is divided into three sections:

- Case File Variables:** A list of variables with two columns: 'Field Name' and 'OriginalName'. The list includes DATA01 through DATA15, TYPE, EVTCODE, PERIOD, YEAR, WAVE, ROCBASE, TRAININGFLAG, FSID, ALTERNATIVE, ACTIVE, ALTER\_CASEID, POSTED, NAME, PHONE, and OTHERID.
- Case Item Layout:** Three lines for defining the case item display. Line 1 contains [DATA01] [DATA02] [DATA03]. Line 2 contains [DATA10], [DATA11] [DATA12] [DATA13]. Line 3 is empty. Each line has a delete button (X).
- Case Info:** A table for managing case details. It has columns: 'Row Order', 'Case File Variable Name', and 'Column Label (ENGLISH)'. The table contains three rows: Row 1 (Name), Row 2 (DATA01, STREETNO), and Row 3 (DATA02, STREET). A 'Clear Grid' button is located to the right of the table.



## 4.4 Data Aggregation

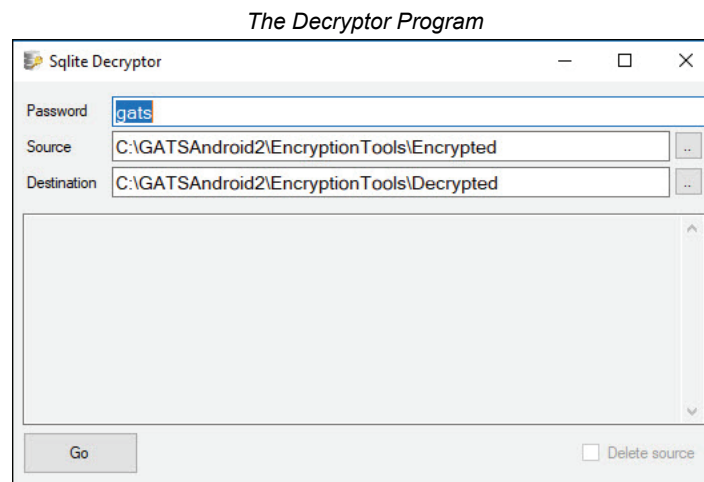
Data Aggregation brings up the data aggregation program and/or one of its suboptions.

### 4.4.1 Database Decryptor

The Database Decryptor allows the developer to set the source location of the encrypted files, the destination for the decrypted files and the password for encryption. The following sections explain each action available to the developer.



The decryptor program will decrypt the selected location files and produce a decrypted version in the destination folder, as seen in the screenshot below. All original source files remain intact unless the checkbox for Delete Source in the decryptor screen is selected.



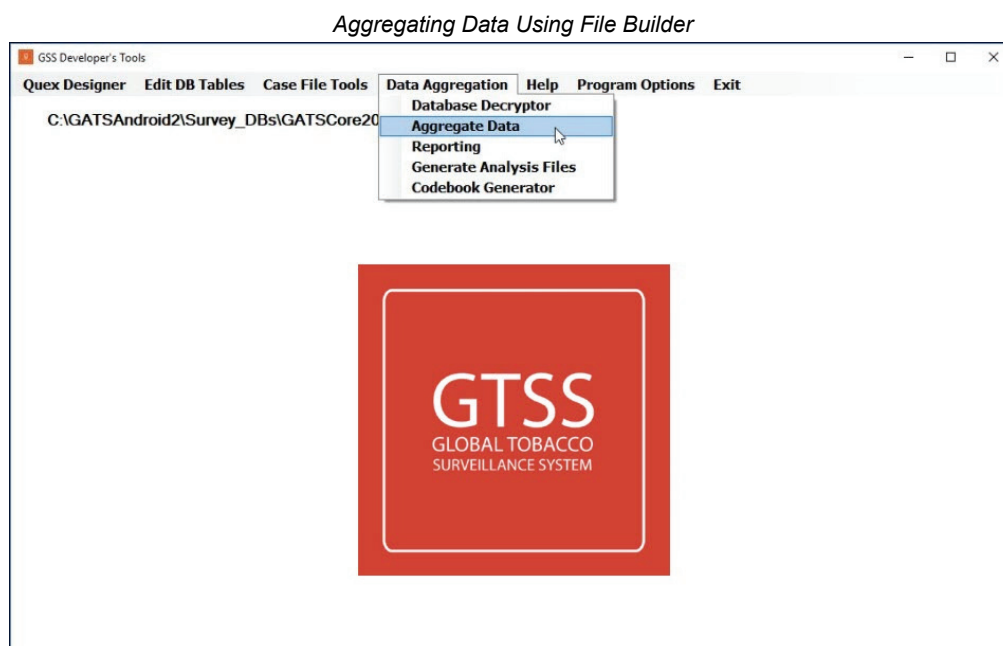
The decrypted files will be used for aggregation, which is described in the next section.



## 4.4.2 Aggregate Data—File Builder Tool

### Overview

File Builder is a program that combines or aggregates FI data files. File Builder should be used for aggregating data from the individual FIs. The File Builder program facilitates collecting and combining the FI files into one cumulative master data file.



File Builder works on SQLite database files, which are supported on both the Android and Windows PC platforms. The GSS software can export SQLite files, capturing all the internal data. These exported files can then be aggregated together to build a master data file that contains all the questionnaire response data, event data, notes, dwelling unit (DU), and address changes for the survey.

The country IT Lead will use File Builder to aggregate data received from handheld devices in the field. Data will be aggregated into a master data file using File Builder.

### **Field Interviewer Files to be Aggregated**

Field Interviewer to next level aggregation:

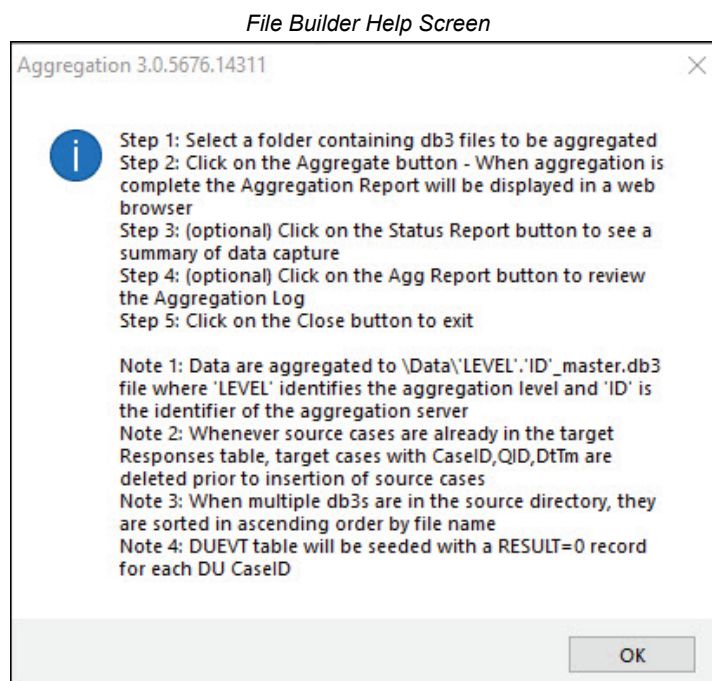
- Field Interviewer runs Export to copy data from the FI device to the export folder.
- Field Interviewer connects handheld device to the internet and the exported file is transmitted to the National Data Center through the cloud (e.g., Dropsync/Dropbox, FTP, etc.).

The data files to be aggregated must exist as SQLite files in an accessible location, such as on the **C:\...\Incoming** folder of the PC running the File Builder tool. Data must be exported before the File Builder data process execution. (You should create a folder like this to store your inbound SQLite files.)

For detailed instructions on using the GSS Export utility, see the *GATS Field Interviewer Manual*.

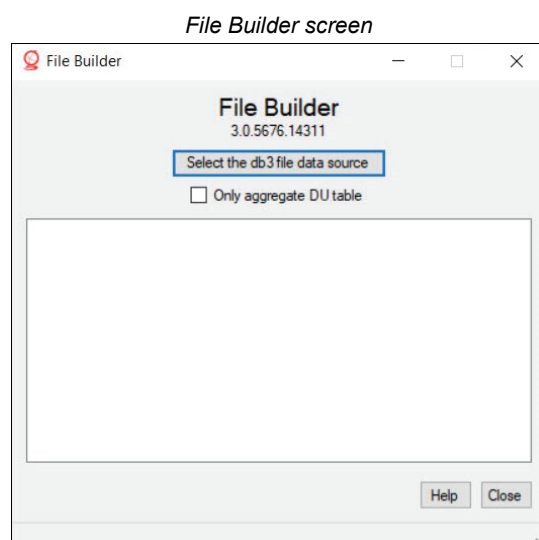
## File Builder Help

A Help screen, available within File Builder, provides some instructions and notes. To access the Help screen, click the **Help** button on the File Builder screen. The following information will appear.

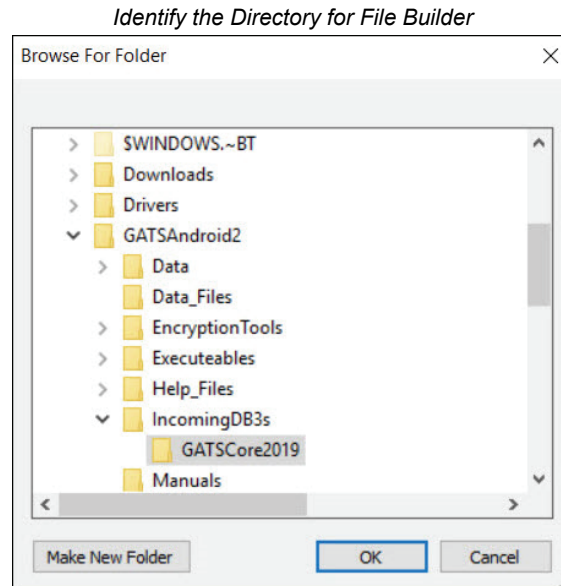


## Aggregating data using File Builder

Click **Select the db3 file data source** as seen in the screen shot below and browse for the directory/folder in which the Field Interviewer data were previously saved. While it is not necessary to maintain the FI data files in a certain folder, they **should not** be stored in **C:\...\Data** folder on the PC where the master aggregated file will be kept.

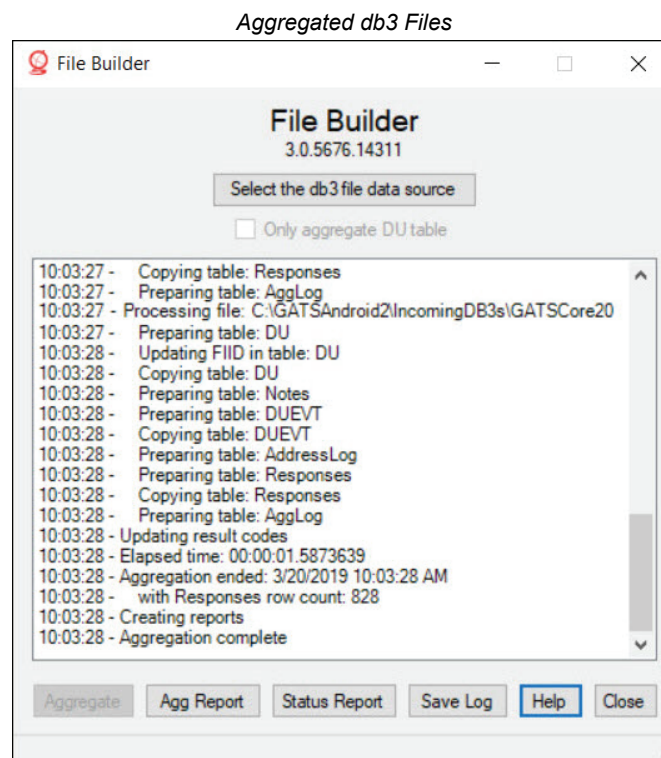


Once the directory containing the db3 data has been identified, click on the folder name, and then click **OK**.



The database files residing in the folder that you selected are then listed on the File Builder screen.

Click the **Aggregate** button. Doing so will aggregate/combine all the files in the folder. A screen is displayed indicating the tables that have been read in from the aggregated db3 files.



## Output Generated by File Builder

Once the data have been aggregated successfully, File Builder can provide several reports, detailed below.

**Aggregation Summary Report:** Directly after the aggregation summary screen is displayed and the aggregation is complete, a summary table showing the log of the aggregation is generated. The contents of the generated file `aggrpt.htm` are displayed using the default Windows HTML editor. A screenshot of the `aggrpt` file is shown on the following page.

The aggregation summary report can be generated by clicking the **Agg Report** button that is displayed after files have been successfully aggregated.

*Aggregation Summary Report Screen (file `aggrpt.htm`)*

GATS Data Aggregation Summary						
Date: 3/20/2019 10:03:28 AM						
AggDateTime	AggTarUnit	AggTarUnitID	AggSrcUnitID	AggTable	AggRowsTotal	AggRowsInserted
3/20/2019 10:03:26 AM	GATS	Core2019	500110	DU	44	44
3/20/2019 10:03:26 AM	GATS	Core2019	500110	DUEVT	4	4
3/20/2019 10:03:26 AM	GATS	Core2019	500110	Responses	223	223
3/20/2019 10:03:27 AM	GATS	Core2019	500120	DU	34	34
3/20/2019 10:03:27 AM	GATS	Core2019	500120	DUEVT	23	23
3/20/2019 10:03:27 AM	GATS	Core2019	500120	Responses	227	227
3/20/2019 10:03:27 AM	GATS	Core2019	500130	DU	34	34
3/20/2019 10:03:27 AM	GATS	Core2019	500130	DUEVT	4	4
3/20/2019 10:03:27 AM	GATS	Core2019	500130	Responses	80	80
3/20/2019 10:03:28 AM	GATS	Core2019	500210	DU	36	36
3/20/2019 10:03:28 AM	GATS	Core2019	500210	DUEVT	28	28
3/20/2019 10:03:28 AM	GATS	Core2019	500210	Responses	298	298

- Note 1: `AggSrcUnitID` is taken from the input `db3` file name
- Note 2: `AggRowsTotal` is the number of rows from the input `db3` file
- Note 3: `AggRowsInserted` is the number of rows inserted from the input `db3` file

**Status Report:** Click the **Status Report** button on the File Builder aggregation screen to generate a status report summarizing the data from the `db3` files (file `statrpt.htm`). A screenshot of the `statrpt.htm` report is shown on the following page.

You can generate the status report by clicking the **Status Report** button that is displayed after files have been successfully aggregated.

This Data Aggregation Summary Report provides information about the results of the aggregation. Each row shows the results for an aggregation unit (typically a Field Interviewer) and a specific database table. The table is denoted in the `AggTable` column, the number of rows in the input table is shown in the `AggRowsTotal` column, and the number of rows inserted from that table is shown in the last column.

AggRowsInserted. The report above shows in the first line that for Field Interviewer 500110 and the DU table, there were 44 rows and all 44 rows were inserted into the master aggregation file.

*Status Report Screen (Statrpt.htm)*

## GATS Household Screening Status

Date: 3/26/2019 8:52:56 AM

Unit	ID	Cases	Unworked	Pending	Final	CompletedInts	Date of Most Recent Data
	999998	0	0	0	0	0	
FI	500210	18	0	11	7	4	11/5/2015 11:01:33 AM
FI	500130	17	0	14	3	3	11/5/2015 11:01:49 AM
FI	500120	17	0	11	6	5	11/5/2015 11:01:29 AM
FI	500110	22	0	20	2	2	11/5/2015 10:52:52 AM
%		100	0	75.7	24.3	18.9	
<b>Totals</b>		74	0	56	18	14	

• Notes: "Cases" is the sum of the "Unworked", "Pending" and "Final" cases. "Pending" includes cases that have been worked, but not finalized (result codes 102,103,104,105,106,108,109). "Final" includes completed interviews and other finalized cases (result codes 200,201,202,203,204,205,206,208,209). "CompletedInts" includes result code 200.

## GATS Individual Questionnaire Status

Date: 3/26/2019 8:52:57 AM

Unit	ID	Cases	Unworked	Pending	Final	CompletedInts	Date of Most Recent Data
	999998	0	0	0	0	0	
FI	500210	4	0	2	2	2	11/5/2015 11:01:00 AM
FI	500130	3	0	3	0	0	
FI	500120	5	0	3	2	0	11/5/2015 11:02:00 AM
FI	500110	2	0	0	2	2	11/5/2015 10:57:14 AM
%		100	0	57.1	42.9	28.6	
<b>Totals</b>		14	0	8	6	4	

• Notes: "Cases" is the sum of the "Unworked", "Pending" and "Final" cases. "Pending" includes cases that have been worked, but not finalized (302,303,304,307,308,309). "Final" includes completed interviews and other finalized cases (result codes 400,401,402,403,404,407,408,409). "CompletedInts" includes result code 400.

The Household Screening Status report is a summary of the status of all cases in the aggregated data file that were reported. For each aggregation unit (typically a Field Interviewer) the total number of cases is reported for that unit in the Cases column. Then the total is broken out into the various status categories. The status columns (Unworked, Pending, Final, and Completed Ints) depend upon the most recent event code for the case. The event codes are defined in the INI table of the CMSDB data base. The typical GATS values are shown below:

INI		
Variable Name	Variable Value	Comment
EvtCodeComplete	200, 400	Event codes that are complete quex
EvtCodeFinal	200, 201, 202, 203, 204, 205, 206, 208, 209, 999, 400, 402, 403, 404, 407, 408, 409	Event codes that are final status codes
EvtCodePending	0, 102, 103, 104, 105, 106, 108, 109, 302, 303, 304, 307, 308, 309, 887	Event codes that are pending status codes

Unworked cases have a most recent event code of 0. Note that the final status includes the cases that are completed interviews. The cases column is the total for a given aggregation unit and should equal the sum of the Unworked, Pending, and Final columns. In the status report for the *Individual Questionnaire*, cases are filtered. Only households that have an event code 200 from household screening (screening interview complete and eligible household member selected) are shown in this table.

### ***File Builder Output Data File—the Master Aggregated db3***

Once the aggregation has successfully completed, File Builder will create a single combined SQLite data file, considered the master data aggregation file, located on the **PC** in the **C:\...\Data** folder. The first time File Builder is run, the master.db3 file is named using the node.id information (e.g., National\_Master.db3). The original db3 file(s) used in the aggregation will reside as it is in its previously saved location.

Browse the **C:\...\Data** folder to find the master db3 file that the aggregation process has created.

### ***Closing File Builder***

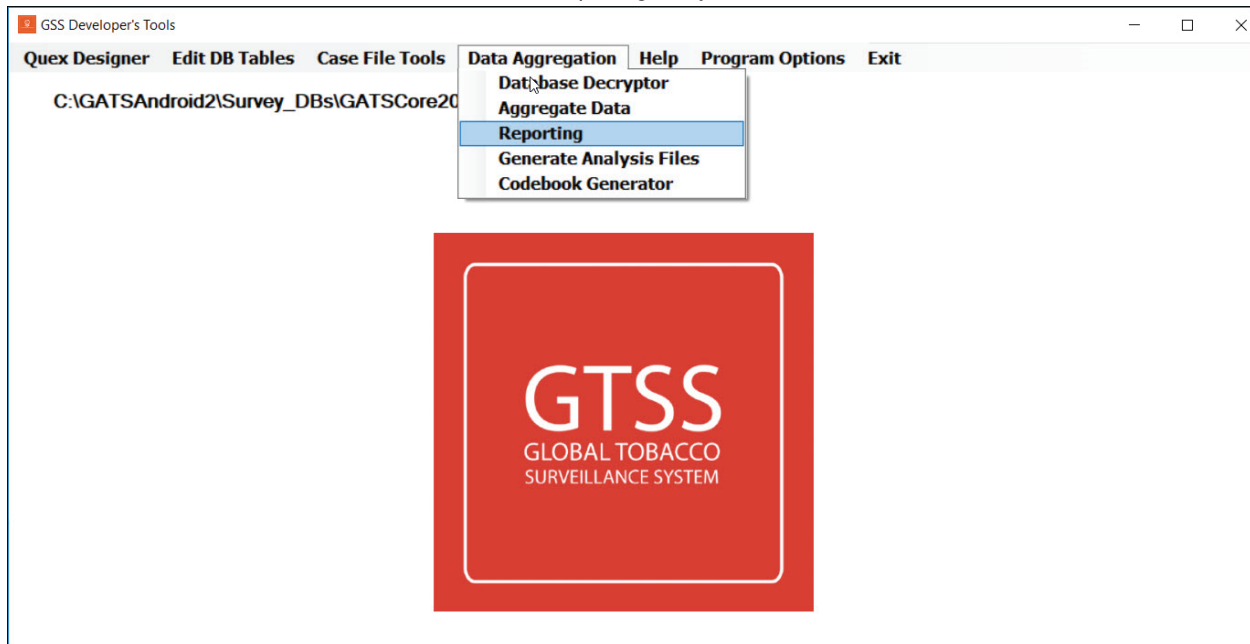
If you click the **Close** button at any time during the aggregation process, the File Builder application closes, and you will exit from the application.

Once the aggregation process has successfully completed and the aggregation and status reports have been generated, click the **Close** button to exit File Builder.

### **4.4.3 Reporting Utility**

The Data Aggregation menu shown in the following screenshot contains a reporting tool that allows users to monitor the content of an aggregated data file for status monitoring and quality control purposes.

## Reporting Utility



The Reporting menu option brings up a standalone utility. This utility allows the GATS manager to create three different status or data reports to view on screen or export to a CSV or text file. The user selects the desired report by tapping on one of the Menu items for reporting—Status HQ and IQ, or Exceptions. The user will be asked to identify or browse the required input files. The user should point the utility to the Master Aggregated db3 file that he/she has made by aggregating the individual database files from the Field Interviewers. The survey db3 files will be used to produce labels and formats for the reports.

## Reporting Utility Screen

Reports Build 3.0.6829.29548

Status HQ and IQ Exceptions Help Save Reports Exit ☐ Allow Training Cases

HQ Case Status IQ Case Status IQ Freqs Exceptions

Event Status — Last Event by FI (from DU table)

FI ID	HRR %	Unworked	200	201	102	202	103	203	109	209	104	204	105	205	106	206	108	208	887	999	299	Totals
500110	100.00	20	2																			22
500120	71.43	10	5		1	1																17
500130	100.00	14	3																			17
500210	50.00	10	4						1			2						1				18
Totals	70.00	54	14		1	1			1			2						1				74

Note: Household Response Rate (HRR) calculation:  $200 / (200 + 102 + 103 + 104 + 108 + 109 + 202 + 203 + 204 + 208 + 209)$



The three reports that are available are the following:

- HQ Case Report—a Survey Event Status Report for the *Household Questionnaire* based on the latest event or Record of Visits code (from the DU table), which details the status of each FI's workload (see the screenshot below for an example).
- IQ Case Report—a Survey Event Status Report for the *Individual Questionnaire* based on the latest event or Record of Visits code (from the DU table), which details the status of each FI's workload.
- Exceptions Report—lists FIs who have exceeded a specified number of visit records for a case but have not coded the case as final. (See the FI Exceptions Cases with No Final Codes and >= Four Event Codes screenshot for an example.) This report is based on the event codes in the aggregated DUEVT table. If an FI has a case that has >= four visit records, but the case is still not in a final status, then that case will be listed in the report. This report will detail separately the cases for the *Household Questionnaire* (HQ) events and the *Individual Questionnaire* (IQ) events. If cases have HQ events but no IQ events, the IQ columns will be blank.

*Survey Event Status Report HQ Case Status Screen*

Reports

Version July 2010

Build 3.0.6829.29548

Status HQ and IQ

Exceptions

Help

Save Reports

Exit

☐ Allow Training Cases

HQ Case Status

IQ Case Status

Exceptions

Event Status -- Last Event by FI (from DU table)

Response Rates				UnWorked	Complt.	BrkOff	Inelig.	None S	Inelig.	R not at home	R Refusal	R Incapacitated	Other Nonresp.	Reopen	Replc'd	Totals					
FI ID	HRR %	PRR %	TRR %		400	302	402	303	401	403	309	409	304	404	307	407	308	408	887	999	
500110	100.00	0.00	0.00	20		2															22
500120	71.43	0.00	0.00	12			1			1			1	1					1		17
500130	100.00	0.00	0.00	17																	17
500210	50.00	0.00	0.00	11		2				3		1				1					18
Totals	70.00	0.00	0.00	60		4	1			4		1		1	1	1			1		74

<

>

Note: Person Response Rate (PRR) calculation: 400 / ( 400+302+304+307+308+309+402+404+407+408+409 )

Total Response Rate (TRR) calculation: HRR x PRR



### FI Exceptions Cases with No Final Codes and >= Four Event Codes Screen

Status HQ and IQ

Exceptions

Help

Save Reports

Exit

☐ Allow Training Cases

HQ Case Status

IQ Case Status

Exceptions

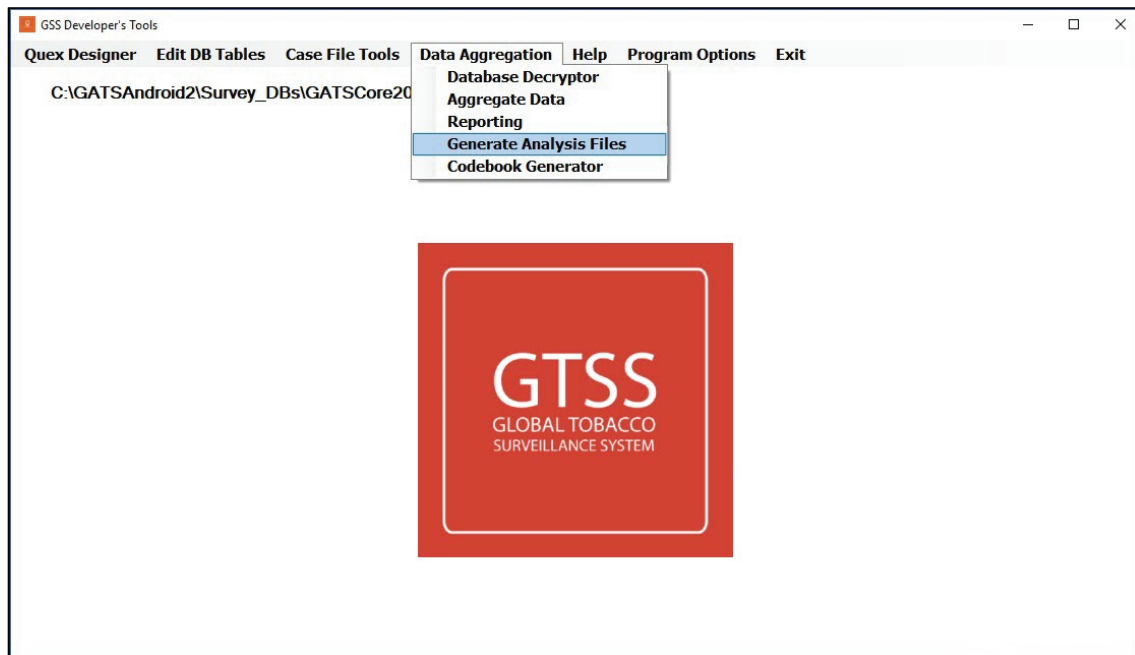
Exception List

Cases with >=4 Result Clodes and no final code

Most recent 5 result codes per Case; HH= household events; IQ= Individual events

FI ID	Case ID	HH Event Count	HH 1	HH 2	HH 3	HH 4	HH 5	IQ Event Count	IQ 1	IQ 2	IQ 3	IQ 4
500120	300400-01	0						6	309 11/5/2015	309 11/5/2015	309 11/5/2015	308 11/5
500210	301450-01	0						6	303 11/5/2015	304 11/5/2015	307 11/5/2015	308 11/5
500210	301460-00	6	103 11/5/2018	104 11/5/2018	105 11/5/2018	105 11/5/2018	109 11/5/2018	0				

### Generate Analysis Files



The system has default values for the location of the required files. The files used in the Generate Analysis Files option include the Master Sample File, the aggregated SQLite database file, and the two questionnaire databases (Survey0.db3 and Survey1.db3). The Master Sample File is the SQLite database used to create the case file for the Field Interviewer devices. Additional fields may be added to this database, but the field names, types, and sizes should not be updated. The field description of fields used by your site should be updated so that the descriptive labels produced by the statistical programs are correct (see the following screenshot).

### Generate Analysis Files Parameters

**Master File Merge**

**Aggregation**

Enter number of iterations to output for variables in loops: 20

Enter location of aggregated file: C:\GATSAndroid2\Survey\_DBs\GATSCore2019\CMSDB.db3

Enter location of GATS\_Survey0.db3: C:\GATSAndroid2\Survey\_DBs\GATSCore2019\Survey0.db3

Enter location of GATS\_Survey1.db3: C:\GATSAndroid2\Survey\_DBs\GATSCore2019\Survey1.db3

Enter Language: ENGLISH

☒ Check here to store List Type questions as numeric

Enter name of CSV file to generate: c:\GATSAndroid2\Data\GATS\_FlatFile.csv

Enter name of SAS program to generate: c:\GATSAndroid2\Data\GATS\_ImportCSVFile.SAS

Enter name of SPSS program to generate: c:\GATSAndroid2\Data\GATS\_ImportCSVFile.SPS

Enter name of STATA program to generate: c:\GATSAndroid2\Data\GATS\_ImportCSVFile.do

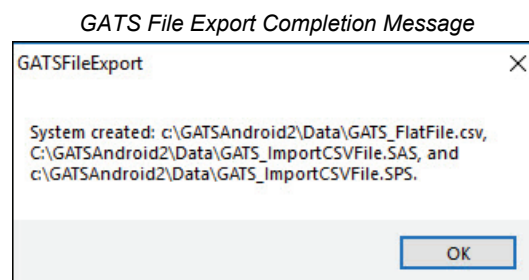
Buttons: Open Databases, Generate File, Exit

Start by confirming the number of iterations/loops is set to 20 and then update the file locations of the master case file, the aggregated file, and the survey questionnaire databases. To change, update the file location or name in the text box or select the "..." button and the system will allow you to select an existing file. Once the file locations and names are correct, select the **Open Databases** button. This will check the file locations and ensure that the system can open the databases. It also gathers information about available languages.

After you have pressed the **Open Databases** button, the databases are open and the option to change or open other databases is not available. You can now select the language for SAS, SPSS, and Stata formatted data to display. At any time, you can select the folder and name of the CSV file, the SAS, the Stata, and the SPSS input programs that are created.

A checkbox allows the user to store list-type questions as numeric. If this option is checked, all list-type questions will be stored as numeric. Note that if a character has been stored in these fields, the SAS, SPSS, and Stata programs will need to be modified to run properly.

Click the **Generate File** button to create the CSV, SAS, SPSS, and Stata files. The CSV will be a comma-delimited file in Unicode (UTF-8) format. This program may take several minutes to run. The hourglass mouse pointer will be visible while the program is running. Once the program is completed, the following message is displayed.



To view the CSV file, a Unicode-enabled text editor such as WordPad is required. The SAS and SPSS files are also in Unicode to ensure that labels and formats are displayed correctly. SAS users should run the program using the SAS Unicode Server. Stata does not support Unicode but will read the comma-delimited file. Text in labels and data that are nonwestern character sets may not display correctly. The statistical programs may need to be updated prior to running them to ensure that country-specific options are set.

#### 4.4.5 Codebook Generator

This utility produces a codebook and example programs in SAS or SPSS. A sample Codebook Generator screen is shown in the screenshot below.

### Sample Codebook Generator Screen

Codebook Generator

Enter number of iterations to output for variables in loops (max: 50)

Enter location of aggregated file

Enter location of GATS\_Survey0.db3

Enter location of GATS\_Survey1.db3

Enter Language

Enter name of the file to generate

Document font

☐ Open document when done

Create Codebook

Exit

Processing variable H02B (list var): -9, 0, REFUSED, 0  
Processing variable H02C (list var): 1, 0, YES, 0  
Processing variable H02C (list var): 2, 0, NO, 0  
Processing variable H02C (list var): -7, 0, DON'T KNOW, 0  
Processing variable H02C (list var): -9, 0, REFUSED, 0  
Processing variable H03 (list var): 1, 0, YES, 0  
Processing variable H03 (list var): 2, 0, NO, 0  
Processing variable H03 (list var): -7, 0, DON'T KNOW, 0  
Processing variable H03 (list var): -9, 0, REFUSED, 0  
Survey 1 appended to codebook in 00:01.43.  
Done.

## 4.5 Help

Help allows users to bring up the following GATS manuals or files:

- *GSS Programmer's Guide*
- *GATS Field Interviewer Manual*
- *GATS Field Supervisor Manual*
- *Data Management Implementation Plan*
- List of Day Codes
- Encrypt/Decrypt String

Help uses the default viewer for portable document format (PDF) files or text files to view these objects.

## 4.6 Program Options

Program Options allows users to set up paths and other required parameters for programs in one location and save them to be passed in to programs as needed. The Developer can add projects by clicking on the Add Project button. This option allows the Developer to associate a set of paths and file names to a specified project.

A sample Setup Parameters screen is shown below:

*Setup Parameters Screen*

Setup Paramters

Project

GATS

▼

Add Project

[1] Home Directory

C:\GATSAndroid2\

...

[2] SD Card Image Path

c:\GATSAndroid2\SD\_Card\_Image\GenRebuild\Programs

...

[3] HQ Path

C:\GATSAndroid2\Survey\_DBs\GATSCore2019\Survey0.db3

...

[4] IQ Path

C:\GATSAndroid2\Survey\_DBs\GATSCore2019\Survey1.db3

...

[5] CMS DB Path

C:\GATSAndroid2\Survey\_DBs\GATSCore2019\CMSDB.db3

...

[6] Case File Path

c:\GATSAndroid2\CaseFiles\CaseFile.txt

...

[7] Help Files Path

c:\GATSAndroid2\Help\_Files\

...

[8] Aggregation Software Path

c:\GATSAndroid2\Bin\

...

[9] Aggregation Data DB Path

c:\GATSAndroid2\Bin\

...

[10] View Data DB Path

c:\GATSAndroid2\Data\

...

[11] Export Data to CSV Path

c:\GATSAndroid2\Data\

...

[12] Transpose Data Survey Path

C:\GATSAndroid2\Survey\_DBs\GATSCore2019\Survey0.mdb

...

[13] Reporting Status HQ/IQ Path

c:\GATSAndroid2\Data\

...

[14] Reporting IQ Freqs DB Path

c:\GATSAndroid2\Data\

...

[15] Reporting IQ Freqs CSV Path

c:\GATSAndroid2\Data\IQ.csv

...

[16] Encryption Password

[17] Screen Title

C:\GATSAndroid2\Survey\_DBs\GATSCore2019

Save

Close Form

## 4.7 Exit

Select to exit the program.



## 5. Files and Structures

---

The CMS and the GSS Engine require an SQL database for operation. RTI has chosen to use the open source SQLite database (see [www.sqlite.org](http://www.sqlite.org)). SQLite runs on most platforms, and specifically it runs on the Android and MS Windows OS platforms required for GSS in the Android environment.

The GSS requires a minimum of two SQLite databases for a given data collection activity, one for the CMS system and one for each GSS questionnaire. These files must be located in the folder on the following path:

`/.../gats/database`

STORAGE should be the name of the Android memory store that is open for shared read write between different applications. Typically, this is the SDCARD store. The folder on the Samsung Galaxy Tab Android device for these files is:

`/.../gats/database`

The files within the database folder must be named according to the following convention:

- The CMS database—CMSDB.db3
- The Survey database—SurveyX.db3

where ProjectName is a character string that the user decides to use as a project name. The database files start with a project name, then the underscore character, and then CMSDB for the CMS and SurveyX, where X is 0, 1, 2, ...  $n$ , and  $n$  is the number of survey forms. The default naming convention for these two database files for a survey with project name ABCD and with only one survey form would be as follows:

- CMS database: CMSDB.db3
- GSS database: Survey0.db3

If there are multiple survey forms, then additional SurveyX.db3 files are needed. In GATS, the files names are:

- CMSDB.db3
- Survey0.db3 (the *Household Questionnaire*)
- Survey1.db3 (the *Individual Questionnaire*)

### 5.1 CMS Database

The SQLite database used by the CMS contains multiple SQL accessible data tables (15) that provide the following (see *Exhibit 5-1* for details on each table):

- Startup parameters and training data for the CMS.  
Tables: INI, DU, TrainingCases.
- Storage tables for control information for the CMS.  
Tables: Delrecs, EvtDef, InterviewInfo, Messages.

- Storage tables for CMS data provided by an FI or CMS system.  
Tables: Addresslog, DuEvt, Tasks, Verinfo, FSTable, PTE.
- Storage tables for GSS survey data provided by the FI (question answers, notes).  
Tables: Responses, Notes.

The following sections detail the design and definitions of the tables used in the databases.

*Exhibit 5-1* details the contents of the CMSDB.db3 database and describes the purpose of each internal table. Detailed information about the columns within each table and the primary key structure is provided in *Exhibits 5-2 through 5-10*, excluding PTE, which is not used in GATS. *Appendix B* provides additional information and some examples of the main data files in the CMSDB of interest to the analyst (Responses, DUEvt, and DU).

#### **Exhibit 5-1. Purpose of Key Internal Tables of CMSDB.db3**

<b>Table Name</b>	<b>Purpose</b>	<b>Required for GATS</b>	<b>Read/Write</b>
Addresslog	Stores address changes and edits made by Field Interviewers.	Yes	R/W
DU	The basic case table where each row represents a case to be worked and tracked by a Field Interviewer. It is loaded with cases at the start of the survey, and cases can be added or removed during survey operations. Contains sample and locator information.	Yes	R/W
DUEvt	Stores event data recorded by Field Interviewers while they work and track their cases.	Yes	R/W
EvtDef	Stores a read-only list of possible events for each form.	Yes	R
INI	Stores lists of startup parameters.	Yes	R/W
Messages	Stores label, error, and information messages in multiple languages.	Yes	R
Notes	Stores note information, by case, as input by Field Interviewers. Serves as a case diary for informal notes about the case.	Yes	R/W
Responses	Stores the response data collected during questionnaire administration in GSS, one row for each question that collects data.	Yes	R/W
TrainingCases	Test cases to be used for training and Field Interviewer practice. Same format and structure as the DU table. ALL training IDs start with XX to distinguish them from live case data in the DU table.	Yes	R/W



**Exhibit 5-2. CMS Database Table: AddressLOG**

Column Name	SQLite Data Type
CASEID	Text
NumCaseID	Text
CREATEDATE	Date
DATA01_O	Text
DATA02_O	Text
DATA03_O	Text
DATA04_O	Text
DATA05_O	Text
DATA06_O	Text
DATA07_O	Text
DATA08_N	Text
DATA09_N	Text
DATA10_N	Text
DATA11_N	Text
DATA12_N	Text
DATA13_N	Text
DATA14_N	Text
DATA15_N	Text
AUTH_CODE	Text
POSTED	Text
POSTDTTM	Date
TZ	Int
TZ1	Int

PRIMARY KEY ('CASEID,' 'CREATEDATE')

**Exhibit 5-3. CMS Database Table: DU**

Column Name	SQLite Data Type
CASEID	Text
NumCaseID	Text
FormNum	Date
ProjectName	Text
SCRNUM	Int
CREATEDATE	Date
TYPE	Text
DATA01_O	Text
DATA02_O	Text
DATA03_O	Text
DATA04_O	Text
DATA05_O	Text
DATA06_O	Text
DATA07_O	Text
DATA08_N	Text
DATA09_N	Text
DATA10_N	Text
DATA11_N	Text
DATA12_N	Text
DATA13_N	Text
DATA14_N	Text
DATA15_N	Text
YEAR	Int
WAVE	Int
ROCBASE	Int
TRAININGFLAG	Text
FSID	Text
ALTERNATIVE	Text
ACTIVE	Text
ALTER_CASEID	Text
PostDTTM	Date
POSTED	Text
Name	Text
Phone	Text
OtherID	Text
_id	INTEGER

\_id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT

**Exhibit 5-4. CMS Database Table: DUEvt**

Column Name	SQLite Data Type
CASEID	Text
NumCaseID	Text
CREATEDATE	Date
CONT_NUM	INT
SCRNUM	Int
EVENTDATE	Date
RESULT	Text
FIID	Text
COMMENTS	Text
HISPBOX	Text
MODIFYDTTM	DATE
AUTO_EVENT	Text
NULIFIED	Text
AUTH_CODE	Text
RECVDTTM	DATE
POSTDTTM	Text
POSTED	Text
TZ	Text
TZ1	Text

PRIMARY KEY ('NumCaseID,' 'CREATEDATE,' 'CONT\_NO,' 'SCRNUM')

**Exhibit 5-5. CMS Database Table: EvtDef**

Column Name	SQLite Data Type
EvtCode	Int
EvtDesc	Text
Status	Real
DTRrequired	Text
AvailableToFI	Text
AllowUpdate	Text
Task	Text
Language	Text

PRIMARY KEY ('EvtCode,' 'Task')

**Exhibit 5-6. CMS Database Table: INI**

Column Name	SQLite Data Type
VariableName	Text
VariableValue	Text
Comment	Text

PRIMARY KEY ('VariableName')

**Exhibit 5-7. CMS Database Table: Messages**

Column Name	SQLite Data Type
PK	Text
MsgID	Text
LanguageID	Text
Message	Text

'PK' INTEGER PRIMARY KEY AUTOINCREMENT

**Exhibit 5-8. CMS Database Table: Notes**

Column Name	SQLite Data Type
CaseID	Text
NumCaseID	Text
QID	Text
Note	Text
DtTm	Date
Version	Text
UserID	Text
PriKey	Integer
Posted	Text
PostedDtTm	Date
TZ	Int
TZ1	Int

'PriKey' INTEGER PRIMARY KEY AUTOINCREMENT

**Exhibit 5-9. CMS Database Table: Responses**

Column Name	SQLite Data Type
CaseID	Text
NumCaseID	Text
QID	Text
Note	Text
DtTm	Date
Version	Text
UserID	Text
PriKey	Integer
Posted	Text
PostedDtTm	Date
TZ	Int
TZ1	Int

'PriKey' INTEGER PRIMARY KEY AUTOINCREMENT

**Exhibit 5-10. CMS Database Table: TrainingCases**

Column Name	SQLite Data Type
CASEID	Text
NumCaseID	Text
FormNum	Date
ProjectName	Text
SCRNUM	Int
CREATEDATE	Date
TYPE	Text
DATA01	Text
DATA02	Text
DATA03	Text
DATA04	Text
DATA05	Text
DATA06	Text
DATA07	Text
DATA08	Text
DATA09	Text
DATA10	Text
DATA11	Text
DATA12	Text
DATA13	Text
DATA14	Text
DATA15	Text
EVTCODE	Text
PERIOD	Date
YEAR	Int
WAVE	Int
ROCBASE	Int
TRAININGFLAG	Text
FSID	Text
ALTERNATIVE	Text
ACTIVE	Text
ALTER_CASEID	Text
PostDTTM	Date
POSTED	Text
Name	Text
Phone	Text
OtherID	Text
_id	INTEGER

\_id INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT

## 5.2 Questionnaire Database Tables

This section discusses the five tables that are included in the questionnaire or survey database.

- **Questions**—Each row is the metadata describing a given question. The link between this table and others is the QID.

- **Texts**—Each row contains the verbatim text (with fills), in given language, for each question in the Questions table that needs text. The link between the Questions table and the QTexts table is the QID column.
- **Answers**—Each row is an option choice, in a given language, for any questions in the Questions table that have option choices (List questions and ATA questions). The link between the Questions table and the Answers table is the QAnswer column in the Questions table and the AID in the Answers table.
- **QxQ**—Each row is the help text for a given question in a given language. The link to other tables for the QxQ table is the QID column.
- **INI**—Each row contains survey setup parameter or data specification. This is where startup and options choices are stored.

Additional details about the purpose of each of these tables are shown in *Exhibit 5-11*. A detailed listing of the columns and formats for each of the questionnaire databases is presented next in *Exhibits 5-12* through *5-16*.

#### Exhibit 5-11. Purpose of Questionnaire Database Tables

Table Name	Purpose	Required for GATS	Read/Write
Answers	Stores answer sets that provide the answer choice options for "list" or "all that apply" questions.	Yes	R
INI	Stores lists of startup parameters.	Yes	R/W
Questions	Contains the main metadata store that provides the data for the GSS questionnaire engine. Each question is a row in this table and provides the logic, validations, etc. for each question. The text for a question is in a separate table, Texts, which has a row for each QID and language.	Yes	R
QxQ	A read-only table of Help information that is provided for all or some of the questions. If QxQ help is available for a given screen, the QxQ menu item is enabled and, when tapped, will show the text provided in this table for a given question.	Optional	R
Texts	Holds the text for questions, one row for a QID and Language. There is a QID row for each language desired.	Yes	R

**Exhibit 5-12. Questionnaire Table: Answers**

Column Name	SQLite Data Type
UID	Integer
AID	Text
Language	Text
ASequence	Int
ACode	Text
AText	Text
AWav	Text

Primary key—UID (AUTOINCREMENT)

**Exhibit 5-13. Questionnaire Table: INI**

Column Name	SQLite Data Type
VariableName	Text
VariableValue	Text
Comment	Text

Primary key—VariableName

**Exhibit 5-14. Questionnaire Table: Questions**

Column Name	SQLite Data Type
QSequence	Real
QIDID	Text
QNext	Text
QLogic	Text
QType	Text
Format	Text
QAnswer	Text
RangeLo	Real
RangeHi	Real
LoopStart	Text
LoopEnd	Text
LoopGoTo	Text
RosterNumber	Int
RosterCol	Int

Primary key—QID

**Exhibit 5-15. Questionnaire Table: Texts**

Column Name	SQLite Data Type
Pkey	Integer
QID	Text
Language	Text
QText	Text
QWav	Text

Primary key—Pkey (AutoIncrement)

**Exhibit 5-16. Questionnaire Table: QxQ**

Column Name	SQLite Data Type
QID	Text
QxQText	Text
QxQWav	Text
Language	Text
PK	Integer

Primary key—PK (AutoIncrement)



## 6. Configuring the GSS for a Handheld Device

---

This chapter provides the GSS user with the information required to prepare the hardware and software systems to use in data collection. Each subsystem is discussed separately, and users may not use all the subsystems.

### 6.1 How to Configure an Android Device

This version of the GSS Case Management System and the GSS Engine is designed to run on an Android platform (Version 4.0 or later). The majority of development and testing have been done on the Samsung Galaxy Tab A device (SM-T280, Android Version 5.1.1). The GSS software is installed using APK via FTP or transferred from PC to the device, and further updates to the databases may be made by manual update.

In the GATS folder are five subfolders for GSS storage as shown in *Exhibit 6-1* below.

#### Exhibit 6-1. Android Folder Structure for GSS within `./gats`

Folder Name	Purpose
backup	Contains backups made by the CMS system
database	Storage locations for the CMSDB.db3 and Survey0.db3, Survey1.db3 files; also where any image files for display would be stored
Downloads	Storage location for files downloaded from the FTP transmission code; typically a case file with new cases for the FI
Exports	Storage location for file exports created by CMS export option
media	Storage for pictures captured using GSS picture capture screens

### 6.2 GSS Files

The files that make up the GSS system fall into several categories and are displayed and annotated below.

#### Programs

YYYYMMDD.apk - GSS program install package

#### Databases

CMSDB.db3 - The SQLite database for the CMS

Survey0.db3 - The SQLite database that defines Survey0

Survey1.db3 - The SQLite database that defines Survey1

Typically, the user will modify only the database files (db3 files) unless changes are made to the various programs. The GSS requires at a minimum two database files for a given survey: one database that contains the data tables for driving the CMS and one that defines a survey and provides the input to the

GSS Engine. If multiple survey forms are used, there will be additional db3 database files, one for each additional survey. For example, if there were three survey forms in a GATS application, an additional file would be required: Survey2.db3. The file names must start with the project name convention and an underscore in order for GSS to find the correct file.

Once these files are copied to the database folder, the user can launch the GSS by running the GSS \_CMS.exe program. This step will launch the CMS system and the other programs as needed. You can create a shortcut for the CMS and place it on the Android Home screen to start the CMS.

### 6.3 How to Configure the Case Management System

The CMS has two startup requirements. First, an initialization file in the CMS database (the CMSDB.db3 file) allows the user to set a number of startup parameters. Second, a file or table that provides the source data for cases must be loaded into the CMS. This second requirement depends upon how data are transferred. The following sections describe how to set up the INI table for the CMS. Additional sections detail how to prepare files or tables for the CMS case loading; one section describes the process for using a Project Web site to manage data movement, and another section describes the process for using memory cards to manage data movement.

The CMS program reads startup parameters from a table in the CMSDB.db3 file when it starts. This is the INI table in the database. It a simple table structure with only three fields:

- VariableName
- VariableValue
- Comment

Each row contains a configuration parameter with a unique ID stored in the Variable Name field or column. The value of the parameter is stored in the Variable Value field, and the Comment field is a text comment that describes the function of the parameter. *Exhibit 6-2* shows the INI parameters and describes their functions.

**Exhibit 6-2. CMS Initialization Table, the INI Table in CMSDB.db3**

Variable Name	Variable Value	Comment
AllowAddressEdit	Yes	Yes or No to allow FI to edit address fields of Cases
AllowCaseTransfer	N	Y or N to allow to transfer cases.
Breakoff	Yes	Flag to control breakoff option (Yes/No) Allow=Yes, Not allow=No
DwellingUnitTable	Yes	Does this project use a Dwelling Unit table (Yes/No)
EvtCodeFinal	200,201,202,203,204,205,206,208,209,999, 400,402,403,404,407,408,409	Event codes that are final status codes
EvtCodePending	0,102,103,104,105,106,108,109,302,303, 304,307,308,309,887	Event codes that are pending status codes

Variable Name	Variable Value	Comment
EvtCodeScreened	200	Event code indicating screening complete and >=1 person selected for more data collection
EvtcodeComplete	200400	Event codes that are complete quex
FIID	555011	FI's six-digit ID
FIName	Mary Field	Name of Field Interviewer, used as a fill in the Q
FTP_Password	*****	FTP login password
FTP_RemoteFolder	InBoundFis	Default folder name on FTP site
FTP_URL	ftp.rti.org	FTP URL
FTP_Username	gats	Login username for FTP site
GovID	123456	Property ID or serial number of the handheld
GridColWidths	120,40,100,45,180,24,48,48,48,48,0,0,0,0,0,48	Pixel widths of Variables that are in CMS grid
GridVars	CaselD,FormNum,EVTCODE,StreetNo,Street,Apt_Num,City,State,ZipCode,County,HC,Box_Num,Rural_Route,Tract_BG,Block,SK,Type	Variables that are in CMS grid
HARDWARE	GALAXY TAB A	Type of handheld device used for survey
Lang0Font	Tahoma	Font used for Lang 0
Lang0label	Eng	Language zero menu button label
Language0	ENGLISH	Name of language to use as initial language for Lang0
LockROCok	YES	Lock ROC so can't add events until HQ case is a 200 code
LoginAttempts	0	Used to track the number of failed login attempts
LoginRequired	No	For internal CMS Yes if want to force login
MessageLanguage	ENGLISH	Language for system Messages
NumLanguages	1	Number of languages used
NumberofSurveys	2	How many different surveys in this project 1 to n
PTEFunction	No	Does this project use the PTE function (Yes/No)
Password	gats	User password required to log in to GSS
PeriodLimit	3/30/2020	Used to set test for unlikely login date time
Qversion	Ver 1 April 2019	Version of the INI file
QxQOption	Yes	Turn on QxQ option: Yes or No
ShowFormNo	YES	Yes or No to show form No in cases grid
TimeZoneOffset	0	Pseudo Time Zone used by CMS to reset PPC
UserName	a	User Name (used by Xmit)

### **6.3.1 Case Files**

To use the CMS (which manages cases), the user must set up the list of cases for the project that will be allocated to the FIs to use on their devices.

### **6.3.2 Setting Up a Case File**

#### **ID Variables**

CASEID (Required field)  
NumCaseID (Required field)  
FIID (Required field)  
FormNum (Required field)  
ProjectName\*  
EvtCode

#### **Sample Variables**

TYPE  
PERIOD  
YEAR  
WAVE  
ALTERNATIVE  
ALTER\_CASID

#### **Locator/Address Variables**

DATA01  
DATA02  
DATA03  
DATA04  
DATA05  
DATA06  
DATA07  
DATA08  
DATA09  
DATA10  
DATA11  
DATA12  
DATA13  
DATA14  
DATA15

### CMS Control Data (not to be changed by users)

SCRNUM  
ROCBASE  
CREATEDATE  
TRAININGFLAG  
FSID  
ACTIVE  
PostDTTM  
POSTED  
Name  
Phone  
OtherID

*Exhibit 6-3* provides a list of the data field names, data type (text, number, date/time), and the function of that field.

**Exhibit 6-3. Data Fields of the Cases Table (CMSDB DU)**

DU		
CASEID	TEXT	"CASEID" TEXT
NumCaseID	TEXT	"NumCaseID" TEXT
FormNum	TEXT	"FormNum" TEXT
ProjectName	TEXT	"ProjectName" TEXT
SCRNUM	INT	"SCRNUM" INT
CREATEDATE	DATE	"CREATEDATE" DATE
TYPE	TEXT	"TYPE" TEXT
DATA01	TEXT	"DATA01" TEXT
DATA02	TEXT	"DATA02" TEXT
DATA03	TEXT	"DATA03" TEXT
DATA04	TEXT	"DATA04" TEXT
DATA05	TEXT	"DATA05" TEXT
DATA06	TEXT	"DATA06" TEXT
DATA07	TEXT	"DATA07" TEXT
DATA08	TEXT	"DATA08" TEXT
DATA09	TEXT	"DATA09" TEXT
DATA10	TEXT	"DATA10" TEXT
DATA11	TEXT	"DATA11" TEXT
DATA12	TEXT	"DATA12" TEXT
DATA13	TEXT	"DATA13" TEXT
DATA14	TEXT	"DATA14" TEXT
DATA15	TEXT	"DATA15" TEXT
EVTCODE	TEXT	"EVTCODE" TEXT
PERIOD	INT	"PERIOD" INT
YEAR	TEXT	"YEAR" TEXT
WAVE	INT	"WAVE" INT
ROCBASE	INT	"ROCBASE" INT
TRAININGFLAG	TEXT	"TRAININGFLAG" TEXT
FSID	INT	"FSID" INT
ALTERNATIVE	TEXT	"ALTERNATIVE" TEXT
ACTIVE	TEXT	"ACTIVE" TEXT
ALTER_CASEID	TEXT	"ALTER_CASEID" TEXT
POSTDTTM	DATE	"POSTDTTM" DATE
POSTED	TEXT	"POSTED" TEXT
NAME	TEXT	"NAME" TEXT
PHONE	TEXT	"PHONE" TEXT
OTHERID	TEXT	"OTHERID" TEXT
_id	INTEGER	"_id" INTEGER NOT NULL PRIMARY KEY AUTOINCREMENT

### 6.3.3 Using an External File to Distribute Cases

The Case File can be read by the FI directly from the Android device. The CMS menu item Load Cases will look in the user initialized folder within the system on the device for a generic case file named CASEFILE.txt. When an FI uses the Load Cases option, the CMS will search for the file. If the CMS finds the file, it will load all cases (rows in the CaseFile table) that match interviewer's FIID (see *Exhibit 6-4*). It will add cases to an FI's CMS grid if the CaseFile table Active field is set to Y. It will remove/hide cases from an FI's device if the CaseFile Active field is set to N. The FIID for a given Android device is defined on the CMS Admin → Settings screen.

**Exhibit 6-4. List of Fields/Variables for CaseFile Table**

Field Name	Contents
FIID	ID of target interview
CaseID	Case ID
NumCase	All numeric case ID must be unique
FormNum	Set to 0
ProjectName	GATS
SCRNum	Set to 0
CreateDate	Set to date record created
DATA01	Address info
DATA02	Address info
DATA03	Address info
DATA04	Address info
DATA05	Address info
DATA06	Address info
DATA07	Address info
DATA08	Address info
DATA09	Address info
DATA10	Address info
DATA11	Address info
DATA12	Address info
DATA13	Address info
DATA14	Address Info
DATA15	Address Info
EvtCode	Set to 0
Period	Set to 1
Year	Set to 1
Wave	Set to 1
ROC_Base	Leave blank
TrainingFlag	Set to N
FSID	Set to ID of Field Supervisor or 1
Alternative	Leave blank
Active	Set to Y to add a case Set to N to remove a case
Alter_CaseID	Leave blank
PostDtTm	Leave blank
Type	Set to Male or Female or Both

## 6.4 How to Configure the GSS

The GSS program, like the CMS, reads in a startup or initialization table at program launch. Many user parameters can be set to start the GSS; these are all stored in the SQLite database for the survey in the table INI. The user can set these parameters using any tools that manipulate the tables in the db3 database. Each distinct form used in a given project has its own SQLite database file and its own INI table. Users most often will use the same parameters across forms. *Exhibit 6-5* details the parameters that are available for the user to set up the GSS and shows the default value or recommended value for GATS.

**Exhibit 6-5. GSS Initialization Table, the INI Table**

Variable Name	Variable Value	Comment
Breakoff	Yes	Flag to control breakoff option (Yes/No)
NumLanguages	1	Number of languages used
Qversion	Version 1.0 April 2011	Version of this INI file
QxQOption	Yes	Turn on QxQ option (Yes/No)
StartupLanguage	1	Language to use Q startup, 0=English, 1=Spanish
SurveyID	Android CORE Individual	Unique ID for this Survey
TestingRndID	No	Used to generate random IDs if = Yes
TextToCaps	Yes	Convert all text entry to Caps (Yes/No)
LangSpecs01	ENGLISH,Tahoma,09	Lang 1: English, font Tahoma, fontsize 9
LangSpecs02x	FRENCH,Tahoma,10	Lang 2: French, font Tahoma, Size 10
LangSpecs03x	SPANISH,Tahoma,11	Lang 3: Spanish, font Tahoma, size 11
DisplayOldData	Yes	
FastForwardOK	Yes	
DisableMainMenu	No	Disable the main menu (Start Screen) if Yes. If no keep the main menu
FastForwardCount	50	Max number of questions to fast forward, there is a limit or GSS may crash

CONFIGURE FTP

## 6.5 Language Customization

The GSS programs allow the use of multiple languages for survey forms; in addition, all of the program screen text and system messages can be set to a specific language. In the CMSDB.db3 file, a Messages table provides translation of messages to a target language. The default language for GSS is English, but users may translate the English messages to their target language. Approximately 450 English messages are used in the GSS programs. These messages are used for labels, column headings, information messages, error messages, and so forth. Each message has a unique ID, a language name, and text. If users want to develop their own set of messages in the target host language, they should add additional rows to the Messages table—one row for each English message, with the same message ID but a new language name. *Exhibit 6-6* shows the layout of the Messages table.

**Exhibit 6-6. Messages Table**

PK	MsgId	LanguageID	Message
7580	ERROR_IN_FILL_MISSING_BRACE	ENGLISH	Error in fill variable missing brace.
7617	MISSING_REQUIRED_PASSWORD	ENGLISH	Missing required password, encryption process aborted.
7639	QUEX_ERROR_BAD_CHILD_COUNT	ENGLISH	An error occurred while calculating the master questionnaire ID. Missing eligible child count. Contact technical support.
7640	QUEX_ERROR_BAD_PARENT_IDENTIFIER	ENGLISH	An error occurred while calculating the master questionnaire ID. Missing or invalid parent identifier. Contact technical support for assistance.
7645	REQUIRED_SETTINGS_CODE_INVALID	ENGLISH	Invalid or missing authorization code.
8102	ERROR_MISSING_TAG	ENGLISH	Error in symbol evaluation, missing closing tag.

The fields are as follows:

- PK: an auto-number field that is the primary key.
- MsgID: a unique ID, within Language, for each message.
- Language ID: identifies the language of the message.
- Message: the verbatim text of the message.

As noted, if users want to change the language, they should add a new row for each message. For example, to add a new message for French, the user will start with message "frmLists001" and add a row with the three fields, as shown below:

```
Frmlists001      FRENCH      French text for this message
```

Typically, users will develop a matching message in the target host language for every English message in the Messages table. GSS looks for messages in the table using the language defined by the CMSDB INI table parameter, MessageLanguage. If GSS cannot find the message in the language specified by MessageLanguage, it looks for that message in English.

Note: *Appendix C* contains frequently asked questions and answers. *Appendix D* provides a list of acronyms used in this manual and their definitions.



## Appendix A: GSS Programming Language

---

This section provides the GSS user with a description of the syntax and usage for the GSS programming language. This language is used in the QLogic block of the metadata that defines a GSS question or screen. The programming logic is available to allow developers to add flow control, variable creation, evaluate expressions, and call-user-supplied subroutines.

### **General Syntax**

The general format of a GSS logic statement is a text sentence ending with a semicolon, for example:

```
If {HH1} == 15 then goto A1;
```

The sentence above is an If Then statement that uses a symbol (the text {HH1}, defines a symbol) and a GoTo statement. GSS allows up to 150 sentences in a given logic block for a question. Each sentence should end with a semicolon.

### **Operators**

The GSS If Then statements support the standard mathematical operators allowed in Java (<, >, <=, >=, ==).

### **Symbols**

GSS supports the concept of symbols. In a regular programming language, these are called variables. GSS maintains an internal table of symbols called the symbol table. Symbols are added to the symbol table automatically as progress is made through an interview, and users can add symbols to the table in logic blocks with a set statement. For example, every time a question is answered, a symbol is created or updated in the symbol table and, hence, is available for users as fill-in question text or in calculations in logic blocks. For example, if you wanted to use a symbol in question text, you would enter the text in the Texts table. If on question A1 you collected a person's first name, then you could use that first name on subsequent questions. The text for a subsequent question might look like this where you use the person's first name to personalize the question:

```
{A1}, can you tell me how old you were on your last birthday?
```

GSS would look up the value of the symbol {A1} and substitute its current value in the text when the question is presented.

A symbol is defined as a variable name surrounded by curly braces, for example {HH1} or {FirstName}. Symbol names must be less than or equal to 20 characters and must be unique within a questionnaire.

Symbols can have a subscript to indicate the instance level at which they exist. Note that if one is inside loops in GSS, then the variables are at instance levels greater than 0. If the subscript is left off, GSS assumes the symbol is at instance level 0.

## Statements

The GSS logic parser supports the following statement types:

- **If Then:** The If Then statement is a standard If Then programming statement that lets conditional logic be applied. Table A-1 shows several examples of the syntax for this statement. If the expression evaluates to True, the statement following the "then" will be executed. If the expression evaluates to False, the statement after the "then" will not be executed. If there are multiple If Then statements in a logic block, they are executed sequentially unless there is a GoTo statement or a Backupto statement that is executed; execution of subsequent logic will cease after the first executed GoTo or Backupto.

- **Set:** The Set statement allows users to create new variables or update the values of variables in the symbol table. The general format is as follows:

```
Set {V1}='Harry';
```

The above statement creates a symbol V1, if none exists, or updates the existing symbol V1 with the value "Harry." Symbols can have a subscript to indicate the "instance" level at which they exist. Note that if one is inside loops in GSS, then the variables are at instance levels greater than 0. If the subscript is left off, GSS assumes the symbol is at instance level 0. The syntax to specify a specific symbol noting its instance level is as follows:

```
Set {V1_0}='Harry';
```

```
Set {V1_3}='Harry';
```

- **GoTo:** The GoTo statement allows users to skip directly to a new question (QID). The syntax of the GoTo is:

```
GoTo XXX;
```

where XXX is the valid name of a QID in the current questionnaire. GSS does not check the validity of the QID until run time, so if the QID does not exist, it will cause a run time error on the device.

- **Backupto:** The BackUpTo statement allows users to back up to an old question (QID). The syntax of the BackUpTo is:

```
BackUpTo XXX;
```

where XXX is the valid name of a QID in the current questionnaire. GSS does not check the validity of the QID until run time, so if the QID does not exist, it will cause a run time error. The BackUpTo statement allows users to code flow control that will back up correctly and invalidate data as it backs up over questions. For example, if you had answered questions Q1, Q2, Q3, and Q4 and then issued on Q4 a BackUpTo Q1 command, then the data for Q3, Q2, and Q1 would be flagged as invalid, showing that it had been backed over.

- **Expression:** A GSS expression is a standard programming Java-style expression. Users can use built-in functions from Java like RND() or Len() or regular math and logic expressions. Expressions can be used in the If Then statement or on the right of set statements. Examples of expressions can be seen in *Exhibit A-1*.

## Exhibit A-1. Examples of Statements

Statement	Examples
BackUpTo	BackUpTo C06a;
If Then	if {C06b} == 888 then goto C06b1; if ({E05} == 1)    ({E05} == 3) then goto E07;
Set	set {RsAge} = {A03}; if {Consent4} == 1 then set {Consent5Text} = '';
Goto	if {C06d} == 888 then goto C06d1;
Call	Call Code_Event;

The BackUpTo statement forces GSS to back up (correctly flagging invalid data as it backs up) to the target screen provided as the one parameter for BackUpTo. The Goto function forces GSS to jump directly to the screen provided as a target; it is not recommended that you use GoTo to jump backwards because it may leave data orphaned as it jumps back over completed questions—jumping forward typically is fine. The Set function causes a symbol to be created and have a value stored in it that can be used later in the program using the {} conventions to call back variables from the symbol store.

- **Call:** The call statement allows users to call user-written or existing system-installed subroutines that can create custom code. If users wish to create callable subroutines, they must code the new routines and generate a new compiled version of GSS with the callable routines.

## Functions

Several Java functions can be used inside the programming language. These are standard Java library functions, and their documentation is covered in any Java manual. The list of functions is shown in *Exhibit A-2*. The name of the function is case sensitive, as shown in the table below. Additional functions can be added if needed by extending the class EvalFunctions.

## Exhibit A-2. Functions Available in QLogic Sentences

String Functions	Math Functions	Miscellaneous Functions	Date/Time Functions
Instr	Sin	Format	DatePart
Rnd	Rnd		Now
Right	Random		[Date]
Mid	Round		Today
Mid1	ConvInt		
Len	Int		
Trim			
Ucase			
LCase			
Left			



## Appendix B: Review of the Main Data Tables

---

This section of this document aims to provide the information and detail needed to work with the GSS data tables once data have been collected in a survey. In GSS all data collected during survey work are stored in a single SQLite database in five tables. These tables are stored in the CMSDB.db3 database file. In GATS, the data from both survey forms are stored within the same table and are distinguished by their different Case IDs and/or Form Numbers. In the GSS three main data tables store data during the course of a survey (others play a minor role [address logs and case notes]). The three main tables are listed below:

- The **DU** table (see *Exhibit B-1*). In GATS, up to two cases or data collection activities will be worked for each dwelling unit (DU): one activity for the screener (HQ) and, if the screener is successful in yielding a person to interview, then a second activity, the individual interview (IQ). The DU table rows contain case identification information, sample information, and geographic locator information (addresses or other information to identify and locate the household). This table contains one row for each case that is to be worked. These rows are loaded into the device as field work progresses to provide the cases that an FI is to work; in other words, to define the survey sample. The "base case" CaseID nnnnnn-00 or case for the HQ interview is loaded at first and if a person is found to be interviewed in the "base case," then a second row is created for that case, by the GSS software, for the IQ interview CaseID nnnnnn-01. It also holds a copy of the most recent event or status code.
- The **Responses** table (see *Exhibit B-2*). This table contains the data that FIs enter as they step through a GSS questionnaire. There is one row in the Responses table for each question asked in a GSS questionnaire. Every time the FI swipes Forward, a new row is written to the Responses table with the data collected on that screen. Every time the FI backs up over a question, the Responses table row for that question is flagged as invalid, marking it as having been backed over. The Responses table also stores internal or system variables that GSS or the GSS programmer output during an interview.
- The **DUEvent** table (see *Exhibit B-3*). This table contains the status or event codes collected during the course of working a case. These codes may be generated automatically during or interview or added by hand by the FI. For more information about the generation and management of event data, see the *GATS Field Interviewer Manual* and *Field Supervisor Manual*.

Two other tables contain additional data. The Notes table holds the FI notes that have been entered and the AddressLogs table holds any address updates that have been made.

The specifications below detail the variables or columns of the three data tables. The DU table has one row for every case where a case is a given data collection activity, either a screener questionnaire or an IQ in GATS. Its layout and the definitions of the columns are shown in *Exhibit B-1*.

## Exhibit B-1. DU Table—Table of Case-Level Records for Each Unique Data Collection Activity

Column Name	Definition	Type	Notes
CaseID	Unique ID for the case, used in all case tables	Text	Must be unique within the entire survey sample
NumCaseID	Alternate ID or 2ndary for the case	Text	Not used in GATS but sometimes used for additional ID information
FormNum	Questionnaire question ID	Text	In GATS 0=Screenner (HQ); 1=Individual (IQ)
ProjectName	Raw data answer to question QID	Text	Used to identify the project within and between countries
EvtCode	Most recent event code for this case	Text	
ScrNum	Number of times that this case has been reassigned	Int	Only used with Web-based case assignment
CreateDate	Date case record/row created	Date	
Data01	Street number of dwelling unit or survey location	Text	All rows shaded gray are locator information that can be adapted country by country
Data02	Name of street	Text	
Data03		Text	
Data04	Box number of dwelling unit	Text	
Data05	Rural route number of dwelling unit	Text	
Data06	Field Interviewer ID number	Text	
Data07			
Data08			
Data09		Text	
Data10	Name of city	Text	
Data11	Name of state	Text	
Data12	ZIP code	Text	
Data13	Name of county	Text	
Data14			
Data15			
Block	Sample Information	Text	
Type	Sample Information	Text	
Period	Sample Information	Int	
Year	Sample Information	Text	
Wave	Sample Information	Int	
ROCBBase	Base code for event data	Int	Only used by Web case assignment
TrainingFlag	Flag to indicate a training case	Text	Y=Training case; N=not training, hence a live case
FSID	ID of Field Supervisor who owns this case	Int	Only used by Web case assignment and transmission
Alternative	Flag to indicate if this case is an alternative case in a pair of cases	Text	Not used in GATS
Active	Flag to indicate if the case is active or not	Text	Y or N; if set to N, case will not display in the CMS listings
Alter_CaseID	ID of paired case	Text	Not used in GATS
PostDtTm	Time stamp when case was xmitted	Date	Only used by Web case assignment and transmission
Posted	Flag to indicate row was transmitted	Text	Only used by Web case assignment and transmission
Name	ID information of people or person within HH	Text	Can be used to store information collected during screening or interview to display in case info
Phone	ID information of people or person within HH	Text	Can be used to store information collected during screening or interview to display in case info
OtherID	ID information of people or person within HH	Text	Can be used to store information collected during screening or interview to display in case info

The Responses table (*Exhibit B-2*) contains question-level data and internal variables generated by GSS or the GSS programmer during an interview. In general, it contains one row of information for each question asked during an interview. Every time an interviewer swipes **Forward** on a screen a row of data is written to the Responses table. In GSS only one data item is stored per screen; if there are multiple data items on a screen, they are concatenated together, separated by the "|" symbol, and stored in the Answer column.

**Exhibit B-2. Responses Table—Table to Store Respondent Answers**

Column Name	Definition	Type	Notes
CaseID	Unique ID for the case	Text	Must be unique within the entire survey sample, same ID as in DU table
NumCaseID	Alternate ID or 2ndary for the case	Text	Must be unique within the entire survey sample, same ID as in DU table
QID	Questionnaire question ID	Text	Must be unique within a questionnaire
Answer	Raw data answer to question QID	Text	Note that GSS collects only one data item per screen
Instance	Indicates level of looping if within a loop in the questionnaire	Int	Range 0 to $n$ , where 0 indicates not in a loop and $> 0$ indicates in a loop
Lang	Code indicating language used on this screen	Int	In the range 0–25
DtTm	Time stamp	Date	Standard MS date time
Version	Version stamp of the questionnaire	Text	Version info comes from text associated with GSS START screen
UserID	Field Interviewer ID number	Text	Must be unique over FIs
Prikey	Auto number field used to number rows	Int	Numbers rows uniquely within a device
Valid	Flag to indicate if data field has been backed over or invalidated	Boolean	True or false
OnHold	Yes/No flag indicating row on hold for transmission	Text	Used only for Web-based transmission
Posted	Yes/No flag indicating xmit status	Text	Used only for Web-based transmission
PostDtTm	Time stamp for xmit	Date	Used only for Web-based transmission
TZ	Time Zone indicators	Int	Not used in GATS
TZ1	Time Zone indicators	Int	Not used in GATS

The DUEvent table (*Exhibit B-3*) contains a row for each event that is recorded for a given case. These events may be generated by the GSS questionnaire or entered by hand by the FI. The event table allows one to track and monitor the status of a case as the FI works it.

**Exhibit B-3. DUEvent Table—Table to Record Status Codes for Data Collection Activities or Events**

Column	Definition	Type	Required Data	Notes
CaseID	Unique ID for the case, linked to DU and Responses tables	Text	Yes	Must be unique within the entire survey sample, same ID as in DU table and Responses table
NumCaseID	Alternate ID or 2ndary for the case	Text	Yes	Must be unique within the entire survey sample, same ID as in DU table
Createdate	Date time event created	Date	Yes	
Cont_No	Contact number, sequential count of events for this CaseID	Int	Yes	
ScrNum	Number of times this CaseID has been reassigned	Int	Yes	Starts at 0
EVENTDATE	Date of event as entered by Field Interviewer	Date	Yes	Standard MS date time variable
RESULT	Event code	Text	Yes	
FIID	ID of interviewer	Text	Yes	ID of interviewer
Comments	Free text for Field Interviewer comments	Text	Yes	Free text for Field Interviewer comments
HISPBOX	Not used in GATS	Int	Yes	Not used in GATS
MODIFYDTTM	Time stamp of last mod to an event	Date	Yes	Standard MS date time variable
AUTO_EVENT	Yes/No flag used to indicate if event is system generated	Text	Yes	Y=system generated event, N=event entered by Field Interviewer
NULLIFIED	Yes/No flag event has been nullified	Text	Yes	Used only for Web-based transmission
AUTH_CODE	For events that require an authorization code, it is stored here if used	Text	Yes	Generally a letter followed by 4 digits, for example, a1234
RECVDTTM	Time stamp for receipt	Date	Yes	Used only for Web-based transmission
PostDTTM	Time stamp for posting	Date	No	Used only for Web-based transmission
POSTED	Yes/No flag to indicate event has been transmitted; Y=transmitted	Text	No	Used only for Web-based transmission
TZ	Time Zone indicators	Int	No	Not used in GATS
TZ1	Time Zone indicators	Int	No	Not used in GATS



*Exhibits B-4 and B-5* show sample data for the DUEvent and Responses tables discussed above. *Exhibit B-4* contains a sample of GSS event data. Here each row represents one event associated with a specific case. In this table we can see that for CaseID 100027-00 we have an event indicating that the screener was completed on 5/25/2008 13:38 and another event indicating the IQ was completed for CaseID 100027-01 on 5/25/2008 14:06. Both of these events are auto events indicating that the event was generated as part of the GSS interview process.

*Exhibit B-5* shows sample response data for a GATS Household Questionnaire (CaseID 100027-00) and a portion of the corresponding Individual Questionnaire (CaseID 100027-01). Most data rows represent the answer to a given GATS question and the QID matches the questionnaire specification for the question name, for example, QIDs HH1, HH2, HH3, etc. In the Responses table, note that each interview starts with a QID=APPSTART row and ends with a QID=SURVEYEXIT row. These rows mark the beginning and end of an interview, and they are generated automatically when GSS starts and ends an interview. Also note that some rows contain items that are not specific questionnaire items. These rows represent GSS internal variables that capture data that are GSS related but not direct answers to questions. For example, the variables/rows selectee, sage, sgender, selename, and randomnum are all variables that come out of the selection process of a household member for interviews that identify the roster row, age, gender, name, and random number associated with the selected person in the roster. Langcalc and Langspoken are variables that indicate the percentages of the screens in language 0 and the language (0 or 1) used for >50% of the screens. In addition, note the field labeled Valid. It contains the value FALSE when a variable item has been backed and, hence, superseded; otherwise, it is TRUE.

#### Exhibit B-4. Sample Event Data

CASEID	Num CaseID	CREATE DATE	CONT_N O	SCRNUM	EVENTDATE	RESULT	FIID	COMMENT TS	HISPBOX	MODIFYDTM	AUTO_EV ENT	NULLIFIE D	AUTH_C ODE	RECVDTT M
100101-00	0	5/15/2008 13:36:47	0	0		0	999001							
100024-01	0	5/25/2008 16:59:37	0	0		0	999002							
100022-00	0	5/25/2008 12:30:18	1	0	5/25/2008 12:30:18	200	999002	Complete Screener 1		5/25/2008 12:30:18	Y	N		
100022-01	0	5/25/2008 12:52:51	1	0	5/25/2008 12:52:51	400	999002	Complete IQ		5/25/2008 12:52:51	Y	N		
100024-00	0	5/25/2008 13:03:22	1	0	5/25/2008 13:03:22	200	999002	Complete Screener 1		5/25/2008 13:03:22	Y	N		
100025-00	0	5/25/2008 13:14:55	1	0	5/25/2008 13:14:55	200	999002	Complete Screener 1		5/25/2008 13:14:55	Y	N		
100026-00	0	5/25/2008 13:31:53	1	0	5/25/2008 13:31:53	200	999002	Complete Screener 1		5/25/2008 13:31:53	Y	N		
100027-00	0	5/25/2008 13:38:45	1	0	5/25/2008 13:38:45	200	999002	Complete Screener 1		5/25/2008 13:38:45	Y	N		
100027-01	0	5/25/2008 14:06:39	1	0	5/25/2008 14:06:39	400	999002	Complete IQ		5/25/2008 14:06:39	Y	N		
100026-01	0	5/25/2008 14:23:55	1	0	5/25/2008 14:23:55	400	999002	Complete IQ		5/25/2008 14:23:55	Y	N		
100033-00	0	5/25/2008 16:59:39	0	0		0	999003							
100035-00	0	5/25/2008 12:31:22	1	0	5/25/2008 12:31:22	200	999003	Complete Screener 1		5/25/2008 12:31:22	Y	N		
100035-01	0	5/25/2008 12:52:18	1	0	5/25/2008 12:52:18	400	999003	Complete IQ		5/25/2008 12:52:18	Y	N		
100041-00	0	5/25/2008 13:01:59	1	0	5/25/2008 13:01:59	200	999003	Complete Screener 1		5/25/2008 13:01:59	Y	N		
100041-01	0	5/25/2008 13:14:53	1	0	5/25/2008 13:14:53	400	999003	Complete IQ		5/25/2008 13:14:53	Y	N		
100039-00	0	5/25/2008 13:23:22	1	0	5/25/2008 13:23:22	200	999003	Complete Screener 1		5/25/2008 13:23:22	Y	N		
100039-01	0	5/25/2008 13:38:15	1	0	5/25/2008 13:38:15	400	999003	Complete IQ		5/25/2008 13:38:15	Y	N		
100038-00	0	5/25/2008 13:43:07	1	0	5/25/2008 13:43:07	200	999003	Complete Screener 1		5/25/2008 13:43:07	Y	N		
100038-01	0	5/25/2008 13:56:35	1	0	5/25/2008 13:56:35	400	999003	Complete IQ		5/25/2008 13:56:35	Y	N		

## Exhibit B-5. Sample Responses Data

CaseID	NumCase	QID	Answer	Instance	Lang	DtTm	Version	Userid	PriKey	Valid	OnHold	Posted	PostDtTm	TZ	TZ1
100027-00	27-00	APPSTART		0	0	5/25/2008 13:35:12	HQ-V 1.11 .05.22	999002	4288	TRUE	N	N			
100027-00	27-00	INTRO1		0	1	5/25/2008 13:35:18	HQ-V 1.11 .05.22	999002	4287	TRUE	N	N			
100027-00	27-00	HH1	5	0	1	5/25/2008 13:35:27	HQ-V 1.11 .05.22	999002	4288	FALSE	N	N			
100027-00	27-00	HH1	4	0	1	5/25/2008 13:36:01	HQ-V 1.11 .05.22	999002	4289	TRUE	N	N			
100027-00	27-00	HH2	2	0	1	5/25/2008 13:36:44	HQ-V 1.11 .05.22	999002	4290	TRUE	N	N			
100027-00	27-00	HH3	1	0	1	5/25/2008 13:36:55	HQ-V 1.11 .05.22	999002	4291	TRUE	N	N			
100027-00	27-00	HH4		0	1	5/25/2008 13:37:04	HQ-V 1.11 .05.22	999002	4292	TRUE	N	N			
100027-00	27-00	HH4A	Person 1	1	1	5/25/2008 13:37:40	HQ-V 1.11 .05.22	999002	4293	TRUE	N	N			
100027-00	27-00	HH4B	31	1	1	5/25/2008 13:38:25	HQ-V 1.11 .05.22	999002	4294	TRUE	N	N			
100027-00	27-00	HH4D	1	1	1	5/25/2008 13:38:28	HQ-V 1.11 .05.22	999002	4295	TRUE	N	N			
100027-00	27-00	HH4E	2	1	1	5/25/2008 13:38:35	HQ-V 1.11 .05.22	999002	4296	TRUE	N	N			
100027-00	27-00	HH4F	1	1	1	5/25/2008 13:38:37	HQ-V 1.11 .05.22	999002	4297	TRUE	N	N			
100027-00	27-00	EDITROSTERINTRO	2	0	1	5/25/2008 13:38:39	HQ-V 1.11 .05.22	999002	4298	TRUE	N	N			
100027-00	27-00	SELECTEE	1	0	1	5/25/2008 13:38:40	HQ-V 1.11 .05.22	999002	4299	TRUE	N	N			
100027-00	27-00	SAGE	31	0	1	5/25/2008 13:38:40	HQ-V 1.11 .05.22	999002	4300	TRUE	N	N			
100027-00	27-00	SGENDER	MALE	0	1	5/25/2008 13:38:40	HQ-V 1.11 .05.22	999002	4301	TRUE	N	N			
100027-00	27-00	SELNAME	Person 1	0	1	5/25/2008 13:38:40	HQ-V 1.11 .05.22	999002	4302	TRUE	N	N			
100027-00	27-00	RANDOMNUM	0.5386011	0	1	5/25/2008 13:38:40	HQ-V 1.11 .05.22	999002	4303	TRUE	N	N			
100027-00	27-00	HH5		0	1	5/25/2008 13:38:43	HQ-V 1.11 .05.22	999002	4304	TRUE	N	N			
100027-00	27-00	LANGCALC	6	0	1	5/25/2008 13:38:45	HQ-V 1.11 .05.22	999002	4305	TRUE	N	N			
100027-00	27-00	LANGSPOKEN	1	0	1	5/25/2008 13:38:45	HQ-V 1.11 .05.22	999002	4306	TRUE	N	N			
100027-00	27-00	SURVEYEXIT		0	1	5/25/2008 13:38:45	HQ-V 1.11 .05.22	999002	4307	TRUE	N	N			
100027-01	27-01	APPSTART		0	0	5/25/2008 13:39:11	GATS IQ VERSI	999002	4308	TRUE	N	N			
100027-01	27-01	CONSENTINDIA		0	0	5/25/2008 13:39:16	GATS IQ VERSI	999002	4309	TRUE	N	N			
100027-01	27-01	CREAD	1	0	0	5/25/2008 13:39:19	GATS IQ VERSI	999002	4310	TRUE	N	N			
100027-01	27-01	COBTAINED	1	0	0	5/25/2008 13:39:20	GATS IQ VERSI	999002	4311	TRUE	N	N			
100027-01	27-01	INTLANG	2	0	0	5/25/2008 13:39:22	GATS IQ VERSI	999002	4312	TRUE	N	N			
100027-01	27-01	A00		0	1	5/25/2008 13:39:24	GATS IQ VERSI	999002	4313	TRUE	N	N			
100027-01	27-01	A01	1	0	1	5/25/2008 13:39:26	GATS IQ VERSI	999002	4314	TRUE	N	N			
100027-01	27-01	A02A	5	0	1	5/25/2008 13:40:49	GATS IQ VERSI	999002	4315	TRUE	N	N			
100027-01	27-01	A02B	1977	0	1	5/25/2008 13:41:19	GATS IQ VERSI	999002	4316	TRUE	N	N			
100027-01	27-01	A04	1	0	1	5/25/2008 13:41:25	GATS IQ VERSI	999002	4317	TRUE	N	N			
100027-01	27-01	A05	5	0	1	5/25/2008 13:41:44	GATS IQ VERSI	999002	4318	TRUE	N	N			
100027-01	27-01	D14	2	0	1	5/25/2008 13:47:51	GATS IQ VERSI	999002	4319	TRUE	N	N			
100027-01	27-01	D16	4	0	1	5/25/2008 13:49:23	GATS IQ VERSI	999002	4320	TRUE	N	N			
100027-01	27-01	EE01	3	0	1	5/25/2008 13:49:44	GATS IQ VERSI	999002	4321	TRUE	N	N			
100027-01	27-01	E04	1	0	1	5/25/2008 13:49:50	GATS IQ VERSI	999002	4322	TRUE	N	N			
100027-01	27-01	E05	3	0	1	5/25/2008 13:49:57	GATS IQ VERSI	999002	4323	TRUE	N	N			
100027-01	27-01	E07	1	0	1	5/25/2008 13:50:24	GATS IQ VERSI	999002	4324	TRUE	N	N			
100027-01	27-01	E08	1	0	1	5/25/2008 13:50:31	GATS IQ VERSI	999002	4325	TRUE	N	N			
100027-01	27-01	E09	2	0	1	5/25/2008 13:50:48	GATS IQ VERSI	999002	4326	TRUE	N	N			
100027-01	27-01	E11	2	0	1	5/25/2008 13:50:59	GATS IQ VERSI	999002	4327	TRUE	N	N			
100027-01	27-01	E13	2	0	1	5/25/2008 13:51:09	GATS IQ VERSI	999002	4328	TRUE	N	N			
100027-01	27-01	E15	2	0	1	5/25/2008 13:51:21	GATS IQ VERSI	999002	4329	TRUE	N	N			
100027-01	27-01	E17	1	0	1	5/25/2008 13:51:41	GATS IQ VERSI	999002	4330	TRUE	N	N			
100027-01	27-01	FB01A	3	0	1	5/25/2008 13:52:07	GATS IQ VERSI	999002	4331	TRUE	N	N			
100027-01	27-01	FB01B	1	0	1	5/25/2008 13:52:13	GATS IQ VERSI	999002	4332	TRUE	N	N			
100027-01	27-01	FB01C	XXXX	0	1	5/25/2008 13:52:30	GATS IQ VERSI	999002	4333	TRUE	N	N			
100027-01	27-01	FB02	5	0	1	5/25/2008 13:52:41	GATS IQ VERSI	999002	4334	TRUE	N	N			
100027-01	27-01	FB03	4	0	1	5/25/2008 13:52:46	GATS IQ VERSI	999002	4335	TRUE	N	N			
100027-01	27-01	FB04	3	0	1	5/25/2008 13:53:08	GATS IQ VERSI	999002	4336	TRUE	N	N			
100027-01	27-01	G01INTRO		0	1	5/25/2008 13:53:12	GATS IQ VERSI	999002	4337	TRUE	N	N			
100027-01	27-01	G01A	2	0	1	5/25/2008 13:53:26	GATS IQ VERSI	999002	4338	TRUE	N	N			
100027-01	27-01	G01B	1	0	1	5/25/2008 13:53:35	GATS IQ VERSI	999002	4339	TRUE	N	N			
100027-01	27-01	G01C	2	0	1	5/25/2008 13:53:41	GATS IQ VERSI	999002	4340	FALSE	N	N			
100027-01	27-01	G01D	7	0	1	5/25/2008 13:53:44	GATS IQ VERSI	999002	4341	TRUE	N	N			
100027-01	27-01	G01E	2	0	1	5/25/2008 13:53:54	GATS IQ VERSI	999002	4342	TRUE	N	N			
100027-01	27-01	G01F	2	0	1	5/25/2008 13:54:15	GATS IQ VERSI	999002	4343	FALSE	N	N			
100027-01	27-01	G01G	1	0	1	5/25/2008 13:54:25	GATS IQ VERSI	999002	4344	TRUE	N	N			
100027-01	27-01	G01H	mouth publicity	0	1	5/25/2008 13:54:58	GATS IQ VERSI	999002	4345	TRUE	N	N			
100027-01	27-01	G02	2	0	1	5/25/2008 13:55:19	GATS IQ VERSI	999002	4346	TRUE	N	N			
100027-01	27-01	G04A	2	0	1	5/25/2008 13:55:52	GATS IQ VERSI	999002	4347	TRUE	N	N			
100027-01	27-01	G04B	2	0	1	5/25/2008 13:56:16	GATS IQ VERSI	999002	4348	TRUE	N	N			
100027-01	27-01	G04C	2	0	1	5/25/2008 13:56:25	GATS IQ VERSI	999002	4349	FALSE	N	N			
100027-01	27-01	G04D	1	0	1	5/25/2008 13:56:30	GATS IQ VERSI	999002	4350	FALSE	N	N			
100027-01	27-01	G04E	2	0	1	5/25/2008 13:56:48	GATS IQ VERSI	999002	4351	TRUE	N	N			
100027-01	27-01	G04F	9	0	1	5/25/2008 13:57:01	GATS IQ VERSI	999002	4352	TRUE	N	N			
100027-01	27-01	G04G	2	0	1	5/25/2008 13:57:13	GATS IQ VERSI	999002	4353	TRUE	N	N			
100027-01	27-01	G04H	1	0	1	5/25/2008 13:57:40	GATS IQ VERSI	999002	4354	TRUE	N	N			
100027-01	27-01	G04I	2	0	1	5/25/2008 13:58:17	GATS IQ VERSI	999002	4355	TRUE	N	N			
100027-01	27-01	G04J	2	0	1	5/25/2008 13:58:19	GATS IQ VERSI	999002	4356	TRUE	N	N			
100027-01	27-01	G04K	1	0	1	5/25/2008 13:58:44	GATS IQ VERSI	999002	4357	TRUE	N	N			
100027-01	27-01	G04L	2	0	1	5/25/2008 13:58:55	GATS IQ VERSI	999002	4358	TRUE	N	N			
100027-01	27-01	G04M	2	0	1	5/25/2008 13:59:12	GATS IQ VERSI	999002	4359	TRUE	N	N			

## Appendix C: Frequently Asked Questions

---

### ***Hardware/Software Requirements***

#### **Other than Windows OS laptops, what additional hardware is required for GATS data collection and consolidation?**

Additional hardware requirements will be country-specific and dependent on the existing country IT infrastructure and the GATS survey operations and data management functions that need to be performed at the country level. The minimum hardware required at the national level for data collection and consolidation is as follows:

- At least one Windows 7 or later machine (standard laptop), 3 GB RAM, 250 GB hard disk
- A backup device (such as USB memory keys or USB external hard drive).

#### **In addition to the GSS survey administration software on the laptops, is additional software (such as database software and software utilities) required for GATS data collection and consolidation?**

Software applications for data aggregation and some basic status reporting for GATS data collection and management operations will be supplied by RTI, the GATS partner for IT services. There are no additional software requirements for countries transferring data via FTP or Dropbox. Countries transferring data via an FTP server will need FTP server software at their national data site to receive FTP transmissions from the field devices. There are free, open source programs available for this function, for example, FileZilla (<http://filezilla-project.org/>).

#### **When will hardware and software support systems for deploying the data-capture devices and devices for aggregating data need to be in place and operational?**

All hardware and software will need to be in place, tested, and operational 30 days before the full implementation survey is undertaken. Testing of all hardware and software should begin as part of the pretest operations and continue up to 1 month before the full survey.

#### **What are the common problems one can face during the questionnaire administration in a field setting using this software?**

In the field, administration issues stem from two main problems—hardware failure and user errors. A small number of hardware failures are inevitable because the data-capture device gets lost, dropped in a river, or undergoes some other unusual event. Therefore, countries will need spare devices to deploy and should have procedures in place to send replacements to interviewers in the field. User errors, on the other hand, are caused by interviewers not properly understanding procedures or how to manipulate the software. The GSS software was chosen for GATS precisely because of its ease of use and relatively low user error rate. In addition, all countries will receive detailed manuals explaining administration procedures and software use. Lastly, GATS partner organizations will collaborate with host countries to provide in-country training to ensure that interviewers and supervisors have the training they need to successfully administer GATS interviews.

#### **What are the advantages of the GSS software that will be used for GATS?**

GSS offers a case management system and data transmission system. It is also strong at managing flow and skips within a questionnaire, thus allowing for a smooth administration. Furthermore, GSS is

user friendly and requires that interviewers have only a minimum amount of computer knowledge to enter data and manage household assignments. GSS is easy to operate and has been modified to work in each country's native language. In GSS the same logic flow and validations are used for all translations of a given questionnaire, hence ensuring consistency.

**What are the advantages of using Android handheld devices to administer the GATS survey versus paper-based administration?**

There are four main advantages to administering the GATS survey using computers that increase data quality instead of using paper-and-pencil administration. First, the computers' software has built-in validation checks to ensure that the interviewers enter valid responses only. If interviewers attempt to enter an out-of-range response, they will immediately get a prompt to choose among the valid answer choices. Second, the computer manages the flow of the questionnaire and seamlessly routes the interviewer to the next question based on the respondent's prior answers. This enhances the flow of the administration of the survey and eliminates problems of skipped questions, questions asked out of order, and respondents erroneously receiving questions that are not logical based on prior answers. Third, data entry with laptop administration is immediate and simplified. Rather than interviewers writing down responses that data entry clerks later enter into a computer, interviewers directly enter answers into the computer during the interview. Eliminating this extra layer of data entry greatly reduces the chances of data entry errors and allows for quicker delivery of the final dataset. Finally, the computer can store translations of the questionnaire into multiple languages, eliminating multiple forms required for different translations when using paper systems. In GSS the same logic flow and validations are used for all languages, ensuring consistency across languages.

***Data Transmission and Data Processing***

**What types of protocols have been developed for transmitting data from the laptop to the National Data Center? What sort of accommodations are there for countries in which frequent online data transmission may be costly or impractical?**

RTI has two basic data transmission protocols to fit a variety of needs and capabilities. RTI will collaborate with the country's IT staff to adapt these basic models to fit each country's specific needs. The two main data transmission protocols include (1) FTP, and (2) Dropbox.

**What sort of computer knowledge and experience will IT staff need to successfully load and deploy the laptops and manage the data aggregation process?**

Prior to pretest implementation, each country's IT staff will be trained to use the hardware and software needed to support the GATS data collection and aggregation process. At a minimum, IT staff at the National Data Center should have familiarity with running MS Windows-based applications in order to use the PC-based suite of development and aggregation tools. In addition, some programming experience is recommended.

### **How often will data from individual laptops need to be transmitted and consolidated?**

Several dimensions influence decisions about how frequently a country should upload data. They are as follows:

- Device capacity—the Android device can process and maintain 100–200 cases with good results, and we have tested loads that simulate 350 cases for the GATS application (a database of 65,000 rows, a 10 MB database).
- Risk management (i.e., protection against data loss)—although in theory, all interviews could be completed before aggregating data, we recommend that data be aggregated to the next level at least every 7 days, regardless of device capacity. Uploading data daily would be ideal, but not necessarily practical. Automatic backups of the data are made to the internal memory every time the FI exits the case management system. But physical damage or loss of the device can occur and result in the loss of data. Therefore, we recommend uploading data, which creates an off-site copy of the data, as often as is practical.
- Survey monitoring and status reporting—the more frequent the transmission of data, the more up to date the survey monitoring reports based on the data will be. However, GATS protocol also includes guidance for a paper-based monitoring protocol to manage survey progress, which can be used in instances where frequent data transmission is not possible.

### **Will data from the laptop machine require any processing for validation or consistency checks?**

As data are collected, the GSS software performs validation at the instrument level per the questionnaire's technical specifications. It should not be necessary to reapply the checks performed; however, review of the data is highly recommended. Data exported from the data-capture device will be ready for immediate review and subsequent processing as it is aggregated. Several GSS applications are available for viewing the raw data and generating simple reports on the aggregated data. Additionally, tools exist for writing the SQLite files to text/ASCII/CSV files so that country staff can perform further review and processing utilizing software such as SAS, SPSS, and Stata.

### **What kind of facility is in place to make sure that data entered do not get corrupted, and what are the data backup specifications/configurations?**

Upon completion of each interview, data are stored in two places on the Android—in the internal memory and in the folder allocated to storage. These data are stored in a SQLite database. This is a standard open source product that has been well tested and well received. Files can be moved from the FI's device to a desktop or laptop and used by standard analysis software. A third backup of these data files is created each time the interviewer transmits data to the National Data Center.

### **What sort of reporting tools will be available to track the performance and progress of the survey for individual interviewers, field teams, PSUs, and/or regions?**

The exact nature of the performance progress reports will depend upon the frequency and method of the country's data transmissions. The database files generated by the aggregate of data from individual devices are readable by a number of different analysis tools. It will be possible to develop computer-based reports with many standard analysis tools like SAS, SPSS, or Stata. Simple status reports will be available as part of the aggregation tools provided by RTI, but host countries may want to adapt

these to develop their own custom reports. RTI has also outlined an example of a paper-based performance tracking report system for countries where data transmission may not be frequent enough to generate up-to-date electronic reports.

## Appendix D: Acronyms List

---

AID	Answers Identification
ASCII	American Standard Code For Information Interchange
ATA	All That Apply
CMS	Case Management System
CMSDB	Case Management System Database
CSV	Comma Separated Values
DU	Dwelling Unit
DUEVT	Dwelling Unit Event Table
FTP	File Transfer Protocol
GATS	Global Adult Tobacco Survey
GB	Gigabytes
GSS	General Survey System
GTSS	Global Tobacco Surveillance System
HQ	Household Questionnaire
HTML	Hypertext Markup Language
IT	Information Technology
OS	Operating System
PC	Personal Computer
PDF	Portable Document Format
QID	Question ID
SAS	Statistical Analysis Software
SPSS	Statistical Package for the Social Sciences
SQL	Structured Query Language
SQLite	SQLite Open Source SQL Database Software or Database Files
URL	Uniform Resource Locator
USB	Universal Serial Bus
WHO	World Health Organization
XMIT	Transmission Program







## Global Adult Tobacco Survey (GATS)