



IHAB-rl Mobile Software v1.4

Documentation

Last updated: March 16, 2020

Source: https://github.com/IHAB-RL/IHAB_MobileSoftware.git

Contents

1	Introduction	1
2	Installation	1
2.1	Requirements	1
2.2	Installation steps	1
3	Usage of the application	2
4	Developer Mode	3
5	Compilation of a questionnaire	3
5.1	Location of questionnaire files	3
5.2	Head	3
5.3	Timer	3
5.4	Title	4
5.5	Questions	4
5.6	Finish	5
5.7	Checkbox grouping and exclusivity	5
5.8	Default items	5
5.9	Comments	6
5.10	Answer formats	7
5.11	Logic	7
6	Data transfer	7
7	License	8

1 Introduction

The application you have obtained is a version of the IHAB EMA system. For standalone use no additional equipment is therefore required except for a computer to facilitate installation and data transfer. The software has been tested on the following devices:

- LG Electronics Nexus 5 (Android 5.1 & 6.0.1)
- Motorola XT1032 (Android 5.1)
- Motorola Moto G2 (Android 6.0)
- Motorola Moto G8+
- Lenovo TB2 X30F (Android 6.0.1)
- ...

This software is still in development. Please feel free to contact us in case of any bugs or difficulties occurring during installation and usage.

2 Installation

2.1 Requirements

The device should be reset to its operating system default (*Settings→Backup&reset→Factory data reset*) prior to the installation. All accounts (*Settings→Accounts*) as well as device administrators (*Settings→Security→Device administrators*) need to be removed, too. On an average size display, the font size should be set to *Large* (*Settings→Display→Font size*). Required for the installation are:

- USB connection between smartphone and computer
- Free software *ADB* (Android Debug Bridge), e.g. “Minimal ADB and Fastboot”
- IHAB mobile .apk installation file
- Questionnaire .xml file
- helpful: free Android file manager, e.g. “Total Commander”

2.2 Installation steps

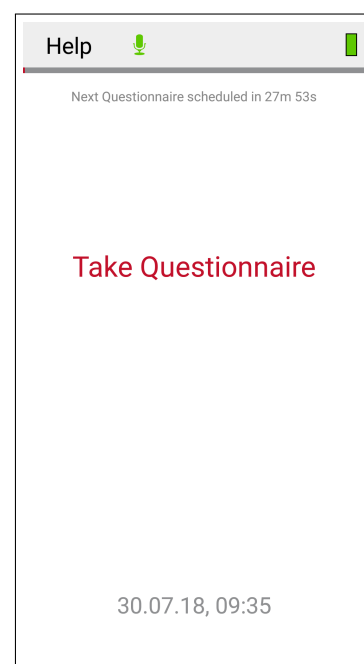
1. Install ADB on the computer and add it to the system path
2. Enable the developer options on the smartphone. How to do this might vary from model to model. The usual procedure is by opening *Settings→About phone*. At the bottom is an entry showing the *Build number*. This needs to be clicked 7 times until a message appears.
3. Go to *Settings→Developer options* and enable *USB Debugging*.
4. Connect the phone to the computer and allow debugging via the dialogue on the phone.

5. The next step is to establish a working connection between the smartphone and the computer. In order to check for a connection, open a console and type in “adb devices”. If everything is okay, ADB should find a device. If not, check your connection. In case the computer does not understand the command, ADB needs to be added to the system path.
6. Now that a connection has been verified, move the console to the directory that holds the .apk file and input “adb install IHAB_Mobile_Standalone.apk”. This should copy and install the application.
7. After the installation is complete, input “adb shell dpm set-device-owner com.fragtest.android.pa/.AdminReceiver”. This makes the application a device owner, which is needed for it to run as a dedicated device.
8. Now input “adb shell appops set android TOAST_WINDOW deny” to keep any information from popping up and confusing the user.
9. In the application settings *Settings*→*Apps*→*IHAB* you need to enable all permissions (Audio input, Memory)
10. Opening the application will throw an error because no questionnaire file is present (however, this step is necessary to build the file system). Using the Windows Explorer, copy your questionnaire file to the *quest* folder within the application directory (*IHAB*). Depending on the device it might be necessary to enable the data transfer protocol MTP on the phone. A downward swipe starting from the top of the display shows the USB protocol which can then be altered. Alternatively you can enter the command “adb push [questionnaire file name] sdcard/ihab/quest”.
11. Restart the phone by entering “adb restart”.

3 Usage of the application

If the installation was successful, the smartphone runs as a dedicated device (or kiosk device), meaning that the user cannot leave the application or make alterations to the operating system. Most buttons are disabled or limited to specific use – including the power button which will only restart the device after a long push.

The menu view (see figure on the right) of the application features a bar at the top with a *Help* button, providing a general explanation of the main answer formats. Next to it is a microphone symbol that can change its color according to whether a wireless connection has been established (green), no connection can be made (gray) or – in this case – standalone mode (blue). On the right hand side is a battery symbol displaying the current state of charge.



The center of the screen features the "Take Questionnaire" button (in case of no major errors present) which can be used to manually start a questionnaire at any given time. Above it is a display of the time until the next questionnaire is triggered. If a questionnaire is prompted the font size increases, accompanied by vibrating alert.

4 Developer Mode

When testing a new questionnaire, it is helpful to disable the KIOSK mode. The easiest way is the *Developer mode*, which can be unlocked by copying a specific file to the main application folder (*IHAB*). The file may be empty and must be named "config" without a formatted ending. On the next start the application will boot into developer mode. The action can be reversed by simply deleting the file from the folder. To enter the in-app preferences menu, simply hold down the "HELP" button in the top left corner for 5 seconds. Inside the preferences, you are free to disable the KIOSK mode as well as change other settings.

5 Compilation of a questionnaire

The questionnaires are generated on the basis of .xml files which can be altered to suit the needs of the experiment. Generally it is a good idea to start from an existing file and make changes, testing them regularly on the smartphone in order to guarantee errorless functionality. In case of formal mistakes (e.g. missing brackets or typos within tags) the application might crash.

5.1 Location of questionnaire files

Questionnaire files are placed in the *quest* folder located in the application directory. The application will automatically sort all questionnaire files alphabetically and pick the uppermost. If you like to use a different questionnaire, you can choose manually from the preferences menu (see 4), but we recommend placing only a single file in the *quest* folder in order to avoid confusion.

5.2 Head

The head consists of two lines that must not be changed, but may become obsolete in future releases.

5.3 Timer

Two different timer settings exist: *interval* and *fixed schedule*. When choosing interval, a `mean` and `deviation` tag and value can be set. The application generates a random interval true to a uniform distribution within the given limits ($\text{mean} \pm \text{deviation}$). This can be beneficial in order to avoid sampling artifacts, e.g. suggesting a questionnaire every 60 minutes to someone who is always listening to the news at that time. If the scheduling interval should not be random, a deviation value of "0" is set. The timer is re-initiated after a questionnaire has been finalized.

```
<timer mean="1800" deviation="300"></timer>
```

If it is desired to suggest a questionnaire at specific times, the `date` tag must be set. The value is entered in form of a String, meaning a time within quotation marks, e.g. "13:47". The system uses the 24h format HH:MM and multiple entries can be entered by concatenating times with a semicolon, e.g. "12:15;17:21;14:41". Times are sorted automatically by the application.

```
<timer date="12:15;13:13;14:41;17:21;8:10;08:12"></timer>
```

5.4 Title

The title tag can be set to personal taste and is formally needed but does not appear in the output result files.

```
<title>
  <text>Exemplary Questionnaire</text>
</title>
```

5.5 Questions

A question consists of an opening `question` tag including some properties, e.g. `id`, `type`, and `filter`. Then follows the label – the actual question. After that comes a suite of answer options, which are defined by a unique `option id`. Although there are no restrictions, it has proven useful in practice to combine `question id` and `#` of the answer using an underscore (the underscore is omitted internally). Below is an example of a question with three answer options:

```
<question id="10104" type="radio" filter="10103_01">
  <label>
    <text>What are you currently doing?</text>
  </label>
  <option id="10104_01">
    <text>Relaxing</text>
  </option>
  <option id="10104_02">
    <text>Eating</text>
  </option>
  <option id="10104_03">
    <text>Paperwork</text>
  </option>
</question>
```

The `question id` serves as a unique question identifier, the `type` sets the answer format (see 5.10) and the `filter` determines the conditions under which the question is included in the current sequence (see 5.11).

5.6 Finish

The last question must be the `finish` page which automatically includes a button that completes the questionnaire and saves the results:

```
<finish>
  <text>Thank you!</text>
</finish>
```

5.7 Checkbox grouping and exclusivity

If only one option out of a group of options might be logically possible (see exemplary options 1-3 below), they can form a group. If one item out of this group is selected, the others are automatically deselected. Multiple groups can exist within one question. An extreme form of this is the `exclusive` tag (option 5), which deselects every other item from the respective list of answers except itself and automatically deselects itself if another item is chosen.

```
<question id="10111" type="checkbox">
  <label>
    <text>Where does sound originate from?</text>
  </label>
  <option id="10111_01" group="1">
    <text>Single person</text>
  </option>
  <option id="10111_02" group="1">
    <text>2-3 persons</text>
  </option>
  <option id="10111_03" group="1">
    <text>4 or more persons</text>
  </option>
  <option id="10111_04">
    <text>Public announcement</text>
  </option>
  <option id="10111_05" condition="exclusive">
    <text>It is quiet.</text>
  </option>
</question>
```

5.8 Default items

An item can be set as preselected default using the `default` tag instead of `option` as shown below:

```
<default id="10109_03">
  <text>Not much</text>
</default>
```

5.9 Comments

Two types of comments are supported by the questionnaire format. Single line comments start with “//”, multiple line comments are parenthesized by “/*” and “*/”.

5.10 Answer formats

The type of answer is set in the opening line of the question tag.

Several answer formats have been implemented:

- **radio**: multiple options, single choice
- **checkbox**: multiple options, multiple choice
- **emoji**: a scale of up to 5 emojis to assess personal feelings
- **text**: free text form
- **sliderFix**: slider with fixed scale and labels, result is answer ID
- **sliderFree**: slider with no scale but labels, result is normalized value
- **finish**: last question, including a “Push to complete” button

5.11 Logic

In order to allow for high specificity and efficiency, questions can be equipped with logical decision parameters that establish whether or not a question will be included in the current suite. The switch is located in the opening line of each question under the tag **filter** in form of a comma-separated concatenation of all included conditions. Two scenarios exist: positive and negative. A question will only be included if **ALL** positive conditions are satisfied and it will not be included as long as at least **ONE** negative condition (indicated by a prefixed “!”) is satisfied (e.g. `filter="10103_02,10103_05,!10111_09"`). Think: positive – necessary criterion, negative – sufficient criterion.

The filter conditions are the option ID’s of answer items from earlier questions of the current questionnaire session. Whenever an answer is chosen by the user, its ID is stored to memory and before a question is displayed, a filter checks whether given conditional ID’s can be found in the memory.

6 Data transfer

For every questionnaire that has been answered, a resulting .xml file is produced which can be found in the *data* folder within the application directory. In order to simplify data transfer we have developed a Matlab interface which can also be used to stop the application (because for safety reasons this is not provided or at least not trivial) and to erase all data from the device. The software is self-explanatory and can be found under GitHub: https://github.com/IHAB-RL/IHAB_DataExtraction.git. Please note that so far it has only been tested under Windows 7.

7 License

Copyright 2018 Institute of Hearing Technology and Audiology, Oldenburg

Licensed under the Apache License, Version 2.0 (the “License”); you may not use this file except in compliance with the License. You may obtain a copy of the License at

<http://www.apache.org/licenses/LICENSE-2.0>

Unless required by applicable law or agreed to in writing, software distributed under the License is distributed on an “AS IS” BASIS, WITHOUT WARRANTIES OR CONDITIONS OF ANY KIND, either express or implied. See the License for the specific language governing permissions and limitations under the License.