Journal of
open research software

SmaRT-CIT: Smartphone App for the Response Time Concealed Information Test with Ready-to-Use Android Deployment

SOFTWARE METAPAPER

GÁSPÁR LUKÁCS

ubiquity press

ABSTRACT

This software is the very first publicly available smartphone application for a valid, scientifically proven lie detection paradigm: the response time Concealed Information Test, which aims to reveal whether or not a certain information detail is known to the tested person. Presently, this application is primarily intended for research purposes, providing a ready, easy-to-use, and completely free option to perform this test in flexible settings (e.g., in large groups or remotely). In the future however, the application could also be field-tested and verified for eventual real-life use by appropriate investigative and other authorities all over the world in various critical scenarios, especially where desktop computers are not available, limited, or impractical.

CORRESPONDING AUTHOR:
Gáspár Lukács
Department of Cognition, Emotion, and Methods in Psychology, University of Vienna, Liebiggasse 5, A-1010 Vienna, Austria
lkcsgaspar@gmail.com

KEYWORDS:
concealed information test; deception detection; lie detection; response time; smartphone

TO CITE THIS ARTICLE:
Lukács G 2021 SmaRT-CIT: Smartphone App for the Response Time Concealed Information Test with Ready-to-Use Android Deployment. *Journal of Open Research Software,* 9: 21. DOI: *https://doi.org/10.5334/jors.341*

(1) OVERVIEW

INTRODUCTION

This software is the first implementation of the Response Time Concealed Information Test (RT-CIT) on smartphones available for general use.

The CIT is a lie detection (or more precisely: memory detection or recognition detection) method that aims to reveal whether or not a certain information detail is known to the tested person. For example, the CIT could include the actual murder weapon (the probe; e.g. "rifle") from a recent murder, and several other weapons (irrelevants; e.g. "knife," and "rope") as the items that would be sequentially presented to a suspect in a random order. When each item must be responded to with a button press, the recognition of the probe (in this case, "rifle") by a guilty person (who is aware of the relevance of that item) will typically result in a slower response to that item than to the irrelevant items. Thereby, based on the probe-irrelevant RT differences, a guilty person can be distinguished from innocent ones.

The RT-CIT paradigm has been many times proven valid (see e.g., [1]), and, importantly, the underlying mechanism of the present software has been separately validated in a recent study, in which it was shown that test results using the smartphone application can be just as reliable as those obtained using conventional desktop computer-based tests [2].

A smartphone-based RT-CIT (SmaRT-CIT [coined by Ailsa E. Millen]) could provide a low-cost portable lie detector that can be used flexibly in various real-life scenarios. Furthermore, in experimental settings, any number of participants may be tested simultaneously by using the application on the participants' own smartphones, thereby sparing time and equipment.

However, the applications used for the above-mentioned study were created only for the specific experiments (e.g., always the same irrelevants, no options to choose any settings, etc.). Hence, they cannot be reused for general purposes. The key mechanism (the CIT test) has therefore been expanded with a comprehensive graphical user interface including customizable items (i.e., allowing probe, irrelevant, and other items to be typed in, or loaded as images, by the user) as well as various additional options and features.

There is currently one comparable software: a web-based RT-CIT implementation for desktop computers [3]. The main difference is that the present application is for smartphones, which, apart from touch responses instead of keypress responses, requires many additional particularities (e.g., screen size and soft keyboard compatibility, true fullscreen, etc.). However, there are platform-independent new features as well: In particular, the possibility to present images, a special option to autofill items and settings using previously stored data (via a designated website), instantly presented detailed summary of outcomes for each of the presented potential probes (after each completed CIT), a "shelf" for previous (full and summary) results, and several translations of the CIT instructions (as well as instruction texts in the code structured to facilitate further translations).

IMPLEMENTATION AND ARCHITECTURE

This application was created using the Ionic (v3) framework, based on TypeScript (ES6), HTML5, and CSS. Following the installation of the necessary software (*Node.js* and the Ionic command-line tools; see *https://ionicframework.com/getting-started*), a new project using this framework can be initiated using the appropriate Ionic command (e.g., "*ionic start newAppName*") in the given operating system's terminal. This creates a basic working template application that may be extended for any specific purpose. In case of the present application, the TypeScript, HTML, and CSS codes specifying the RT-CIT and its surrounding graphical user interface were added to the template in the relevant files are under the src/pages/home and src/providers folders. The home directory contains the general interface and basic mechanics of the application, while the providers directory contains various subroutines (mainly the CIT mechanism in cit.ts, the item generation code in itemgen.ts, and all instruction texts in all languages in the translations.ts). All functions and key variables are explained in comments throughout the code (e.g., how to predefine probes or how item sequences are randomized). Finally, using another Ionic command (e.g. "*ionic cordova build android*"), executable files can be built from the source code for various native mobile as well as desktop platforms, such as Android, iOS, Windows, and web (i.e., browser-based). Detailed instructions are available in the official documentation of Ionic (*https://ionicframework.com/docs*).

For future custom modifications, a copy of the source code can be obtained (from the GitHub repository, via cloning, direct download, etc.) and altered in the relevant files (related to the RT-CIT and the interface) described above. Subsequently, the new application can again be built for any of the supported platforms, and the resulting executable file can finally be given to users for easy installation and use.

However, since coding and compilation require time, expertise, and appropriate tools, the built application has been made available for Android (which covers the vast majority of smartphones in the world [4]) via Google Play: *https://play.google.com/store/apps/details?id=io.gaspar.citapp*. Users can thereby easily install and use the application without any technical knowledge or any other restrictions.

INTERFACE

A video clip (ca. 2 min) that showcases the use of the graphical user interface is available via *https://osf.io/fc5xp/* (or *https://youtu.be/YUuzQy7Jmu0*). A short explanation about the most important details of the CIT along with an example is available at the DEMO option in the dropdown menu (upper-left corner), but there is also a more detailed online documentation at *https://github.com/gasparl/citapp_mobile/#The-CIT-Versions*).

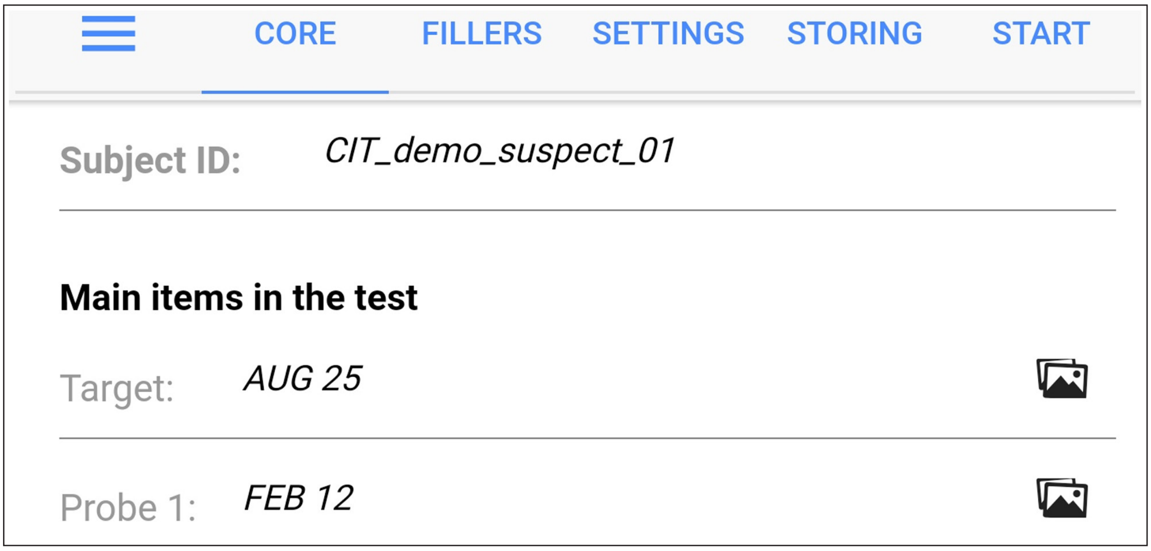
The key part of the graphical interface is the CORE tab, where the user can enter the subject ID and the main items to be used in the test (***Figure 1***).

To note, the so-called "irrelevants" are not designated as such in the interface. Instead, there are five "probes": Probe 1, Probe 2, Probe 3, Probe 4, and Probe 5. The real probe (the item suspected to be recognized by the given examinee) may be given as any of these potential probes (1-5). In the end the results will be evaluated in a way that each of the five potential probes are assessed separately as "real probe" versus the rest of the probes that are for this purpose designated as "irrelevants". The reason for this is that the single "real" probe is not always known in advance: For example, in case of an upcoming terrorist attack, a suspected terrorist may be shown several potential (suspected) locations for the attack, and the one item with the slowest responses may be designated as the "real probe" after the test is completed [5].

The default settings under the rest of the tabs can be considered optimal and may be left unchanged for most purposes (except for instruction language, which should normally be changed, under the SETTINGS tab, to the examinee's native language, if possible).

The CIT can be started under the START tab, after which appropriate instructions will be displayed for the examinee, and eventually the task itself will consist of sequentially displayed items that require touch responses at the lower left or lower right parts of the screen [2] (***Figure 2***).

At the end of the test there are no instructions or button, but only the text "Test completed." (in the selected language). This is to prevent subjects to see the results without permission. To see the results,



Subject ID: *CIT_demo_suspect_01*

Main items in the test

Target: *AUG 25*

Probe 1: *FEB 12*

Figure 1 Example screenshot of a displayed item and the response buttons.

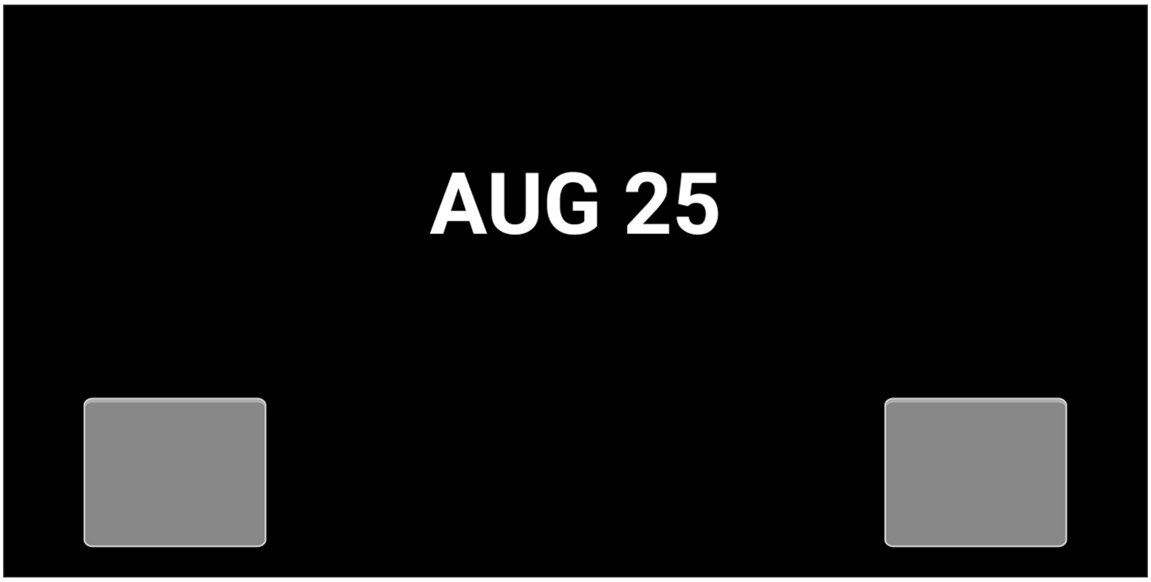


Figure 2 Example screenshot of a displayed item and the response buttons.

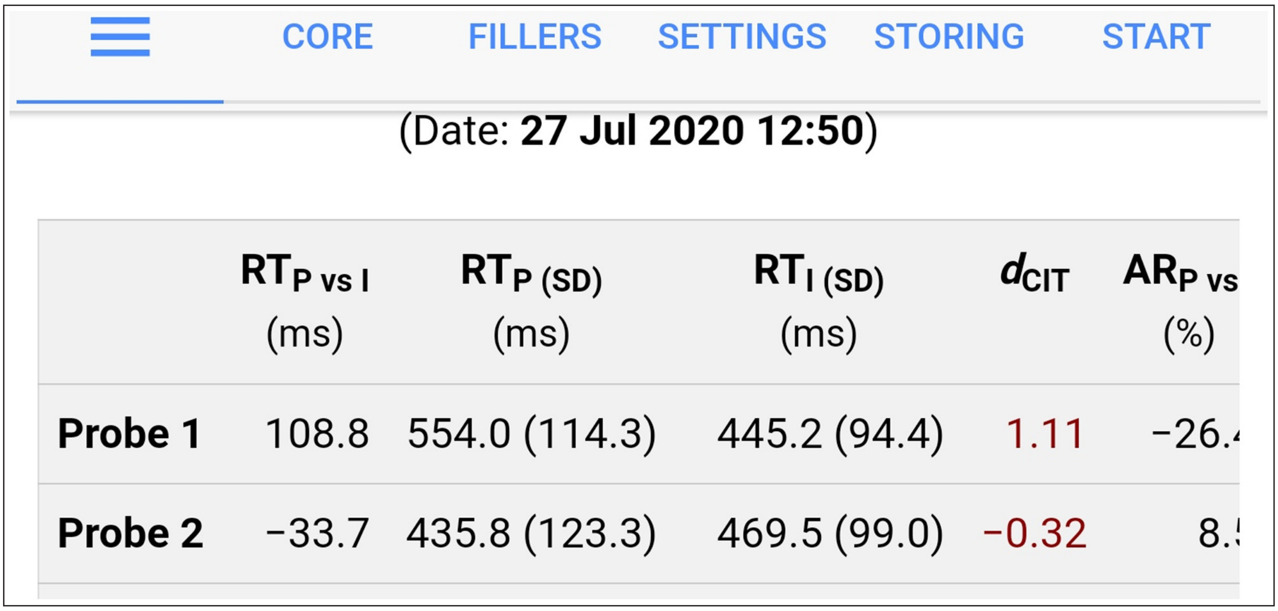
the experimenter should swipe right on this text. The summary results will then be immediately available within the application, including detailed description of each variable and the potential interpretation of the numbers (such as the likelihood of the tested person having recognized any of the given items; ***Figure 3***). The summaries of all completed tests will be permanently available (unless deleted) in a list under the RESULTS menu in the application. The full results file (with each recorded response) is saved on the phone.

As explicitely stated in the application's documentation (under Terms of Use and Privacy), to keep track of very basic usage statistics, after each completed test, the application automatically attempts to forward the following three anonymous details to be saved in a private database: (a) the date (year, month, day) when the test was completed, (b) the language in which the test was completed, and (c) the country code for the approximate location of the testing (based on IP address – but the IP address itself is not saved). If the application is always used offline only, this data will never be forwarded. (A simple way to completely disable this, before building a new instance of the application, is to remove or otherwise disable the *send_stat* function in the *home.ts* file.)

EXPERIMENTAL EXAMPLES

To provide an example including experimental procedure, resulting data and its analysis (as well as to verify the essential integrity of the entire finalized application), an illustrative experiment was conducted on a small convenience sample (student volunteers at the University of Vienna, participating for course credit). Altogether 13 individual CITs were performed (by 7 participants, with 6 performing two CITs), each with a single unique probe.

The probe was selected as either a personally relevant item to the participant (e.g., home country, or any other personal detail that the participant voluntarily gave; simulating guilty suspect), or a random item (e.g., a random country name; simulating an innocent suspect). Corresponding irrelevants (e.g., other random country names; also the "target" irrelevant) were selected by the experimenter, carefully matching all items in general characteristics (e.g., item length) so that a person not aware of probe's relevance would be unlikely to see any substantial difference between probe and irrelevant items. Participants were asked to try conceal their knowledge about the relevant item and recognizing it among the other items. Otherwise, they were not informed about the details of the CIT before the end of the experiment. The settings (e.g., CIT version, instruction language) varied by each test (to verify that they work as expected). Details about each individual test are available in a spreadsheet, along with full results files per test, under the "SmaRTCIT" folder at https://osf.io/fjvna/. (Before testing, each participant was warned about and consented to anonymously sharing these details and data.)



(Date: **27 Jul 2020 12:50**)

	$RT_{P\ vs\ I}$ (ms)	$RT_{P\ (SD)}$ (ms)	$RT_{I\ (SD)}$ (ms)	d_{CIT}	$AR_{P\ vs}$ (%)
Probe 1	108.8	554.0 (114.3)	445.2 (94.4)	1.11	−26.
Probe 2	−33.7	435.8 (123.3)	469.5 (99.0)	−0.32	8.

Figure 3 Example screenshot of summary results. This screenshot was taken of a test completed on July 27, 2020, at 12:50 (local time). The screenshot only captures a part of the actual page: Within the application, scrolling vertically/horizontally would allow looking at all the information. For example, here is an excerpt, taken from the explanatory text below the table, relevant for the present example screenshot: "Response times (for all correct responses, in milliseconds): probe RT mean minus irrelevant RT mean ($RT_{P\ vs\ I}$), probe RT mean and standard deviation ($RT_{P\ (SD)}$), irrelevant RT mean and standard deviation ($RT_{I\ (SD)}$), uncorrected Cohen's d between all probe and all irrelevant RTs (d_{CIT}). Accuracy rates (ratio of correct responses compared to incorrect and too slow responses, in percentage): probe accuracy rate minus irrelevant accuracy rate ($AR_{P\ vs\ I}$), probe accuracy rate (AR_P), irrelevant accuracy rate (AR_I)."

In this experiment, the subject ID always followed the format of, for example, *citapp_01a_p1_gu*. This records three important details: the experiment title "citapp"; the subject number and test "01a" (first participant, first test); the suspected probe number "p1" (e.g., home country was entered as "Probe 1"); and the experimental condition, in this case simulated guilt or innocence, "gu" (guilty). The sixth participant's second test, with suspected but in fact random (non-personal) probe (simulating innocence) given as Probe 4, would be given the subject ID *citapp_06b_p4_in*. The information from the subject ID can then be extracted using a script for automatic analysis, which thereby detects and calculates results for each given suspected probe and compares the main results (typically: probe-irrelevant differences) between the experimental conditions (e.g., a *t*-test between guilty and innocent, in this case).

After the subject ID and other details and settings were entered and selected, the participants started the CIT, and could perform the task following the instructions in the application without any additional help. In the end, using the corresponding functionality in the application, the results data files were forwarded to the experimenter's email address.

The R script for the data analysis (with short explanatory inline comments), is also available under the "SmaRTCIT" folder at https://osf.io/fjvna/. Despite the low sample size, probe-irrelevant RT mean differences were, in line with all previous research, significantly larger in case of "guilty" participants, with 65 ms, 95% CI [15, 125] (*M*±*SD* = 66±47 ms vs. 1±15 ms), *U* = 39, *p* = .008, *d* = 1.79, 95% *CI* [0.45, 3.09] (note: parametric *t*-test is not suitable due to the low sample size, but its result would also be significant, *p* = .010), and diagnostic accuracy far above chance, AUC = .929, 95% CI [.792, 1]. Outcomes (e.g. probe mean RT, probe-irrelevant differences, etc.) obtained using the R script were identical with those depicted in the results summary within the application (see the example in ***Figure 3***). The script may be adjusted for the analysis of the results of any future experiments that use this software.

In this experiment, participants were taken one by one. However, in future experiments using this software, any number of participants could be tested simultaneously, or even remotely. In such a case, it can be helpful to store the items and settings to be used in the given experiment. This can be done via https://gasparl.github.io/citapp_storage/.

For example, if we wanted to explore whether response time limit makes any difference, two separate instances could be stored. In both cases, all settings and items could be the same (e.g., the probe could be the name of the city where the testing takes place, while other, random city names would provide irrelevants), except for subject ID and response time limit. The response time limits, for the two different stored instances, could be, for example, 1000 ms and 1500 ms: They could be given as *Identifiers* the texts "RTlimitEXP1a" and "RTlimitEXP1b" (passwords are not necessary) and as *Subject IDs* "rt_lim_exp1a" and "rt_lim_exp1b."

Subsequently, at the beginning of the testing session, the experimenter may simply ask, for example (assuming the location to be a regular classroom), that those in every odd row enter the Identifier "RTlimitEXP1a," while those in every even row enter "RTlimitEXP1b," under the STORAGE tab, and then press the button "LOAD SETTINGS." (Ideally, participants should have downloaded the application before the session, to save time.) Successful loading is indicated by the message "Data loaded from database." This means that settings were filled with the data specified via the online storage website. At this point, participants could also be asked to add, to the Subject ID, their actual subject number (as indicated, for example, on their individual informed consent form), and/or their age and gender (e.g. "rt_lim_exp1a_54_19_2", where 54 is participant number, 19 is age, and 2 means female), so that these details may be subsequently extracted from the data. The condition in any case is indicated in the automatically loaded subject ID (here: "a" or "b" at the end).

Participants could then proceed with the CIT, and in the end they could be just asked to forward the data by themselves (swipe right on the ending text; press GET FULL DATA button, press CHOOSE APPLICATION button, send email to the experimenter's email address). A "default email" may also be specified in the online storage: In that case, when choosing an email application for forwarding the data (within the CIT application), this email address will be automatically added as "to:" addressee.

The same procedure may be used analogously for testing participants remotely: They are given the instruction via email or a dedicated website, and at the end of the test they send the results to the experimenter.

QUALITY CONTROL

As noted above, the application's CIT mechanism itself has been validated and proven reliable in a dedicated study [2], while the integrity of the surrounding user interface and other new features have been tested in the small-sample experiment described above.

To retest the essence of the application, testers may, for example, enter their own personal names (e.g., a forename) as one of the potential probe items (e.g., "Probe 4"), and other, similar names as the rest of the items. After completing the test, the probe-irrelevant differences calculated for the actual probe (here: "Probe 4," the tester's own name) should be substantially larger than for the other items. (However, since the CIT is not infallible, a more certain way for a quick verification is to make deliberately slow responses to one of the items.) Finally, the summary results may be compared with results obtained using, for example, the provided R script (as explained above), and output files can be verified manually (or again via R code checks) to ensure they contain the expected data as described in the documentation (e.g., all columns present, 162 trials per block in the "enhanced" CIT, 108 in the "standard" CIT, etc.).

(2) AVAILABILITY

OPERATING SYSTEM

Executable file is available for Android 4.4+ OS. (Source code allows builds for other platforms as well, e.g., iOS and Windows.)

PROGRAMMING LANGUAGE

TypeScript, HTML, CSS

ADDITIONAL SYSTEM REQUIREMENTS

Approx. 5.0 MB disk space.

DEPENDENCIES

None.

LIST OF CONTRIBUTORS

None.

SOFTWARE LOCATION

Archive (e.g. institutional repository, general repository) (required – please see instructions on journal website for depositing archive copy of software in a suitable repository)

Name: Zenodo
Persistent identifier: *https://doi.org/10.5281/zenodo.3960310*
Licence: BSD-2-Clause License
Publisher: Gáspár Lukács
Version published: v1.0.1
Date published: 25/07/20

Code repository (e.g. SourceForge, GitHub etc.) (required)

Name: GitHub
Identifier: *https://github.com/gasparl/citapp_mobile*
Licence: BSD-2-Clause License
Date published: 06/05/21

Emulation environment (if appropriate)

Name: none
Identifier: none
Licence: none
Date published: none

LANGUAGE

The language of repository, documentation, and all texts in software are in English, except for the additional translations for the test instructions.

(3) REUSE POTENTIAL

The present application provides a readily available tool for the independent and flexible use of various versions of the RT-CIT [6, 7]. Furthermore, as also demonstrated above, this application allows using the participants' own smartphones in experiments, and thereby data can be easily collected remotely or in supervised settings with larger groups (e.g., dozens or even hundreds of participants), requiring nothing but an empty room for the participants (e.g. a classroom or an auditorium [2]). This way, entire studies could be conducted within half an hour with very little preparation and no equipment costs. (In high-stakes or real-life scenarios however, using an application on the examinee's own smartphone would warrant scrutiny, since the person may malignantly prepare and install, in advance, a modified version that always returns results that imply innocence. Therefore, when there is any suspicion of such countermeasures, using the examinee's smartphone should be generally avoided, or at least it should be ascertained that the installed application was provided by the examiners or other proper authorities.)

As described above (under "Implementation and architecture"), via the source code, all internal settings and structure can be modified as well – which is especially convenient for remote testing, where no direct supervision is possible. For example, to run a study, researchers may want to specify new default settings and prevent any changes by disabling some of the input fields or by starting the CIT task (consent screen) as soon as the application has loaded. At the end of the test, the user may be automatically redirected to a URL to receive compensation, and the results too may be automatically saved on a server via a PHP connection (for which examples are included in the code).

Contact information for inquires about contributions and support are available, along with detailed technical documentation, on the main page of the GitHub source code repository. Contributions are always welcome, for example to add further CIT versions, more setting options, new features (e.g. graphic depiction of outcomes and recognition detection likelihood), or simply to make amendments to the layout. Before making any changes, developers should propose it via opening an "issue" on GitHub or sending an email to the owner of this repository. Eventual code changes will happen via pull requests on GitHub. Those without programming skills can contribute too with translations of the instructions into further languages, or by giving constructive feedback and suggestions.

ACKNOWLEDGEMENTS

Many thanks to Dorota Goc and Claudia "Kiki" Kawai for the Polish and German translations, respectively.

FUNDING STATEMENT

Gáspár Lukács has been funded by the OeAW Post-DocTrack Program. This funding source had no role or involvement regarding the present software.

COMPETING INTERESTS

The author has no competing interests to declare.

AUTHOR AFFILIATION

Gáspár Lukács *orcid.org/0000-0001-9401-4830*
Department of Cognition, Emotion, and Methods in Psychology, University of Vienna, Liebiggasse 5, A-1010 Vienna, Austria

REFERENCES

1. **Suchotzki K, Verschuere B, Van Bockstaele B, Ben-Shakhar G, Crombez G.** Lying takes time: A meta-analysis on reaction time measures of deception. *Psychol Bull*. 2017; 143: 428–453. DOI: *https://doi.org/10.1037/bul0000087*
2. **Lukács G, Kleinberg B, Kunzi M, Ansorge U.** Response Time Concealed Information Test on Smartphones. *Collabra: Psychology*. 2020; 6: 4. DOI: *https://doi.org/10.1525/collabra.255*
3. **Lukács G.** CITapp - a response time-based Concealed Information Test lie detector web application. *Journal of Open Source Software*. 2019; 4: 1179. DOI: *https://doi.org/10.21105/joss.01179*
4. Mobile operating system – Wikipedia [Internet]. [cited 2021 May 5]. Available from: *https://en.wikipedia.org/wiki/Mobile_operating_system#Market_share*.
5. **Meixner JB, Rosenfeld JP.** A mock terrorism application of the P300-based concealed information test. *Psychophysiology*. 2011; 48; 149–154. DOI: *https://doi.org/10.1111/j.1469-8986.2010.01050.x*
6. **Lukács G, Kleinberg B, Verschuere B.** Familiarity-related fillers improve the validity of reaction time-based memory detection. *Journal of Applied Research in Memory and Cognition*. 2017; 6: 295–305. DOI: *https://doi.org/10.1016/j.jarmac.2017.01.013*
7. **Lukács G, Ansorge U.** Information leakage in the response time-based Concealed Information Test. *Appl Cognit Psychol*. 2019; 33: 1178–1196. DOI: *https://doi.org/10.1002/acp.3565*

TO CITE THIS ARTICLE:
Lukács G 2021 SmaRT-CIT: Smartphone App for the Response Time Concealed Information Test with Ready-to-Use Android Deployment. *Journal of Open Research Software,* 9: 21. DOI: *https://doi.org/10.5334/jors.341*

Submitted: 27 July 2020 Accepted: 07 July 2021 Published: 21 July 2021

Journal of Open Research Software is a peer-reviewed open access journal published by Ubiquity Press.