Pocket Lab App (PoLA) Documentation

(ver. 13.03.2024)

Introduction

The Pocket Lab App (PoLA) is a research tool designed to streamline the process of conducting experimental research outside traditional laboratory settings. Free, open-source, and user-friendly, PoLA operates on Android devices to facilitate experiment flow, time management, self-report collection, and data logging. Its integration with PoLA's Helpful Assistant (HelA), an artificial intelligence component, makes protocol creation and revision as simple as providing natural language instructions.

Installation

1. Visit https://lkacz.github.io/pocketlabapp/ to download the PoLA .apk file.
2. Allow installation from unknown sources in your device's settings.
3. Install the .apk file on your Android device.

Getting Started

Upon installation, launch PoLA and familiarize yourself with the interface running demo and tutorial protocols. The app requires uploading a study-specific protocol formatted as a .txt file to guide participants through the experiment.

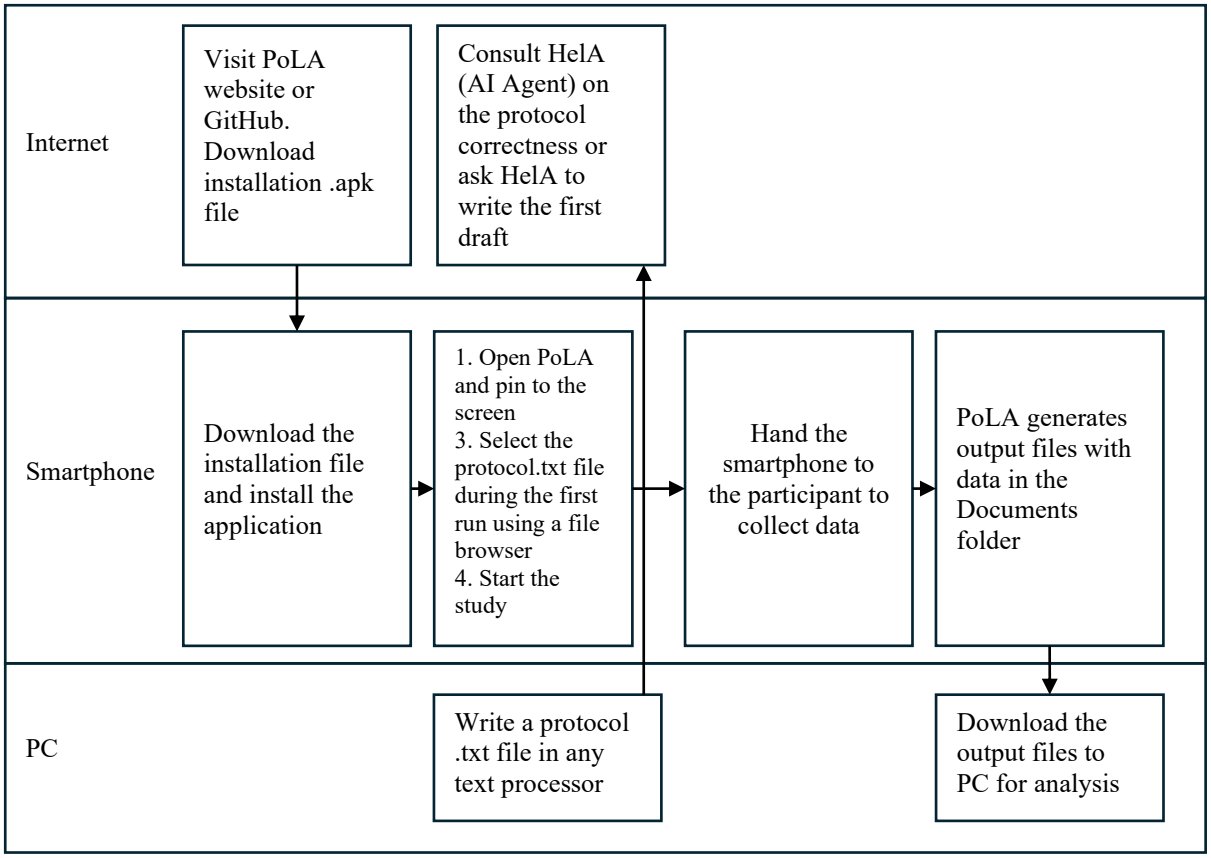


Figure. **Flowchart for Data Collection and Analysis Using PoLA**

Note. The user downloads the installation file from the internet and the protocol.txt file from a PC. The application generates outputs in the Documents folder, which are downloaded to the PC for integration and statistical analysis. The user can use HelA, PoLA's Helpful Assistant, an AI agent that reviews and writes PoLA protocols.

Creating a Protocol

A protocol is a sequence of instructions stored in a .txt file. Each line in the protocol corresponds to a command that PoLA executes. Protocols can be created in any text editor and easily revised or fully developed using natural language with HelA.

Commands Overview

PoLA uses a concise set of commands for various functionalities, including displaying instructions (INSTRUCTION), setting timers (TIMER), collecting self-reports (SCALE, MULTISCALE, INPUTFIELD), and logging data (LOG). The syntax for each command is straightforward, with semicolon-separated elements.

Uploading the Protocol

1. Connect your device to a computer via USB.
2. Transfer the .txt protocol file to the device's storage.
3. In PoLA, select 'Load Protocol' and navigate to the file.

Running an Experiment

1. Start the Session: Launch PoLA on the laboratory smartphone and load the protocol.
2. Participant Interaction: Hand the device to the participant to follow the instructions.
3. Data Collection: PoLA will automatically save responses and log data as the participant progresses through the protocol.
4. End of Session: Retrieve the device, and the collected data will be available for analysis in specified output formats.

Output and Data Analysis

Upon completion of the protocol, PoLA generates output files in both .csv and .xlsx formats, stored under *PoLA_Data* and *PoLA_Backup* folders in the device's Documents directory. These files contain time-stamped participant responses and additional information, facilitating easy and quick data analysis.

Limitations and Considerations

- Requires direct participant-researcher interaction for protocol initiation.
- Designed for moderate complexity experiments; may not support highly specialized tasks.
- Data aggregation across participants requires manual editing.
- The interface and documentation are currently available only in English.
- The AI functionality is experimental and may produce false suggestions that seem correct.

Future Directions

The development team is committed to continuous improvement, with future updates potentially including more complex instructions, diverse self-report options, branching logic, graphical customization, and leveraging smartphone sensors. Community involvement is highly encouraged to ensure PoLA evolves in ways that most benefit the research community.

Developer Information

Lukasz Kaczmarek, PhD., Adam Mickiewicz University, AMU Psychophysiology Laboratory: Gaming & Streaming, Email: lkacz@amu.edu.pl
For inquiries, collaborations, or feedback regarding PoLA and HelA, please contact Dr. Kaczmarek directly via the email provided.

Commands

1. STUDY_ID
Format: STUDY_ID;STUDY_ID_TEXT
Function: This command assigns an identifier to the study, which will be used as a prefix for filenames to help organize data.
Example: STUDY_ID;MyFirstStudy
Usage: It helps distinguish data files from different studies or protocols, especially useful when conducting multiple studies on the same device.

2. SCALE
Format: SCALE;HEADER_TEXT;BODY_TEXT;ITEM_TEXT;RESPONSE1_TEXT;...;RESPONSE9_TEXT
Function: Displays a single question with up to nine response options.
Example: SCALE;Emotions;Rate your current emotional state;Calmness;Very Low;Low;Moderate;High;Very High
Usage: It's ideal for collecting single-item self-report measures from participants like mood, satisfaction, or symptom severity.

3. MULTISCALE
Format: MULTISCALE;HEADER_TEXT;BODY_TEXT;[ITEM1_TEXT;ITEM2_TEXT;...];RESPONSE1_TEXT;...;RESPONSE9_TEXT
Function: Similar to SCALE, but allows presenting multiple items with the same set of response options.
Example: MULTISCALE;Daily Activities;How satisfied are you with the following activities today?;[Exercise;Work;Social Interaction];Not at all;Somewhat;Moderately;Very;Extremely
Usage: Useful for evaluating multiple aspects of a participant's experience or behavior using a consistent rating scale.

4. RANDOMIZED_MULTISCALE
Format: RANDOMIZED_MULTISCALE;HEADER_TEXT;BODY_TEXT;[ITEM1_TEXT;ITEM2_TEXT;...];RESPONSE1_TEXT;...;RESPONSE9_TEXT
Function: Functions like MULTISCALE, but items are presented in a random order each time.
Example: RANDOMIZED_MULTISCALE;Dietary Choices;Rate how frequently you consumed the following items this week;[Fruits;Vegetables;Fast Food];Never;Rarely;Sometimes;Often;Always
Usage: Employed to reduce order effects when presenting multiple items to participants.

5. TIMER
Format: TIMER;HEADER_TEXT;BODY_TEXT;BUTTON_TEXT;NumberOfSeconds
Function: Displays a message and a countdown timer. When the timer expires, a button appears to continue.
Example: TIMER;Break Time;Take a short break. The next session will start shortly.;Continue;300
Usage: Useful for scheduling breaks or controlled time intervals between experiment phases.

6. INPUTFIELD
Format: INPUTFIELD;HEADER_TEXT;BODY_TEXT;BUTTON_TEXT;INPUTFIELD#1;...;INPUTFIELD#10
Function: Displays a form with up to 10 input fields for open-ended responses.
Example: INPUTFIELD;Demographics;Please provide the following information.;Submit;Age;Gender;Education Level;Employment Status

Usage: Collects textual data from participants, including demographic information or open feedback.

7. INSTRUCTION
Format: INSTRUCTION;HEADER_TEXT;BODY_TEXT;BUTTON_TEXT
Function: Shows static instruction or information message with a button to proceed.
Example: INSTRUCTION;Welcome!;Thank you for participating in this study. When you're ready, please press 'Continue' to start.;Continue
Usage: Presents instructions, consent information, or other text to participants before or during the experiment.

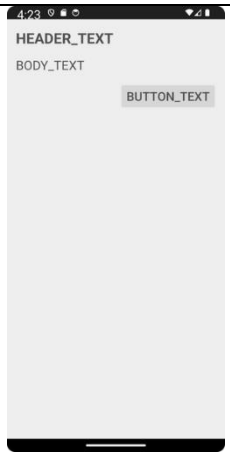
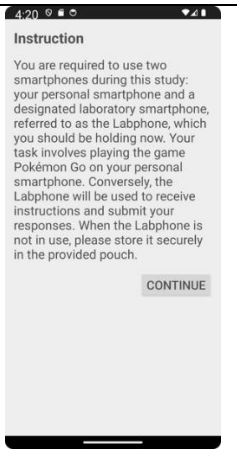
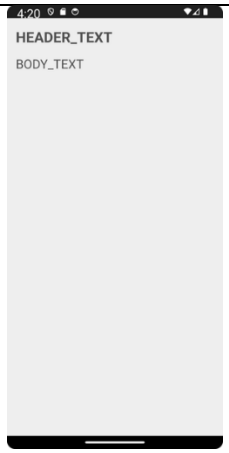
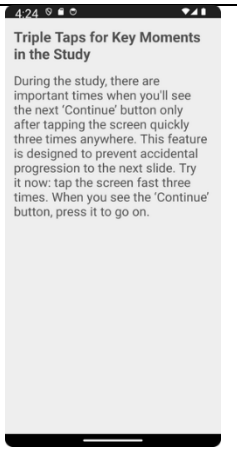
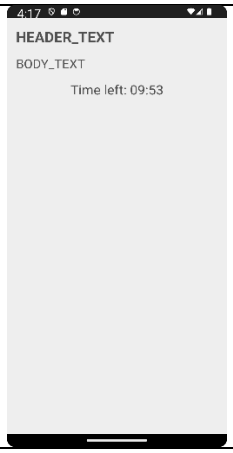
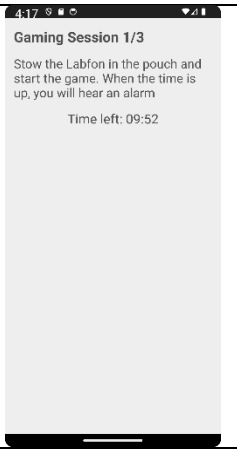
8. TAP_INSTRUCTION
Format: TAP_INSTRUCTION;HEADER_TEXT;BODY_TEXT;BUTTON_TEXT
Function: Same as INSTRUCTION, but requires a triple tap to reveal the continue button.
Example: TAP_INSTRUCTION;Attention Check;To ensure you're paying attention, triple-tap anywhere on the screen to continue.;Continue
Usage: Ensures participants are actively engaged by requiring a specific interaction before proceeding.

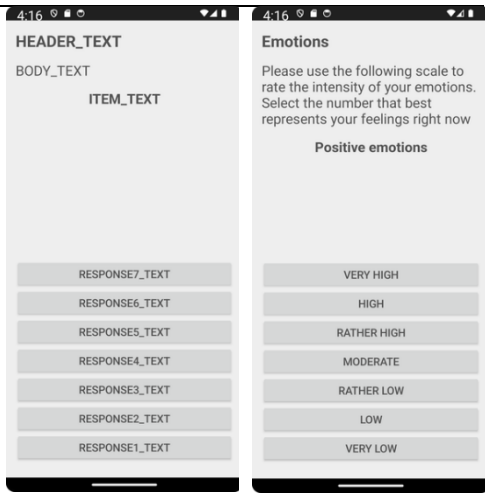
9. LOG
Format: LOG;Text
Function: Logs a text note in the output file when reached in the protocol.
Example: LOG;Starting Section 2
Usage: Useful for marking different phases or sections of the study within the output data for easier analysis.

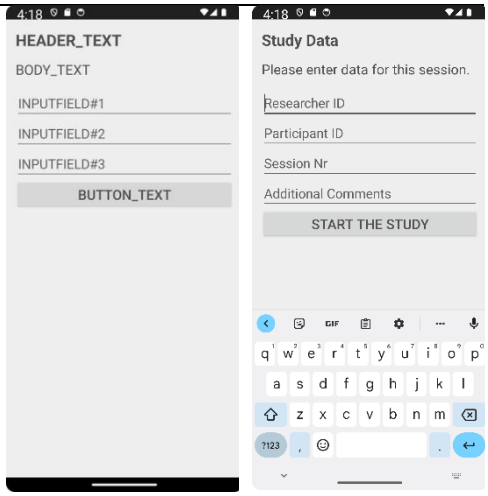
10. END
Format: END
Function: Signals the end of the protocol. Commands placed after this command will not be executed.
Example: END
Usage: Can be used to prematurely terminate the protocol or to ensure that a specific set of commands is the last to run.

Comments
Format and Function: Text not starting with a command identifier is ignored, serving as comments within the protocol file.
Examples: "// This is a comment", "--- Section divider"
Usage: Organizes and annotates the protocol file for better readability and documentation purposes.

Table X. PoLA's commands

Command / format / example	Function	Layout	Example
	Procedure flow		
INSTRUCTION INSTRUCTION;HEADER_TEXT;BODY_TEXT;BUTTON_TEXT INSTRUCTION;Instruction;You are required… ;CONTINUE	Displays header, body, and a button with text. Waits until the participant reads the text and taps the button. Used to present information to the participants (e.g., instructions).		
TAP_INSTRUCTION TAP_INSTRUCTION;HEADER_TEXT;BODY_TEXT;BUTTON_TEXT TAP_INSTRUCTION;Triple Taps for Key Moments in the Study;During the study…;CONTINUE	The same as INSTRUCTION but involves an particular action (triple tap) before the button appears, to ascertain that the user does not skip this slide unintentionally. The button is hidden until the user triple-taps the screen. Used for time-sensitive moments, e.g. before setting the timer or while synchronizing action of two participants. The participant must be instructed on to handled this slide.		
TIMER TIMER;HeaderText;BodyText;ButtonText;Seconds TIMER;Session 1;Walk through the park for the next 10 minutes;Stop alarm;600	This allows researchers to schedule auditory and vibratory alarms. This feature prompts participants to complete assessments or engage in specific behaviors at predetermined intervals.		
	Surveys		

SCALE SCALE;HEADER_TEXT;BODY_TEXT;ITEM_TEXT;RESPONSE1_TEXT;RESPONSE2_TEXT;(up to RESPONSE9_TEXT) SCALE;Arousal Scale;Please report your arousal level right now;Arousal;Very low;Low;Medium;High;Very high	Displays a header, scale introduction, item, and up to nine labeled response buttons. It asks a single question.	
MULTISCALE MULTISCALE;HEADER;INTRO;[ITEM1,ITEM2,ITEM3,ITEM4];LABEL#1;LABEL#2;(up to LABEL#9) MULTISCALE;AFFECT SCALE;Please report your affect levels;[Valence,Arousal,Approach-avoidance motivation]Very low;Low;Medium;High;Very high	The same as SCALE but includes a list of items rather than a single item. The items are presented in the listed order.	
RANDOMIZED_MULTISCALE RANDOMIZED_MULTISCALE;HEADER;INTRO;[ITEM1,ITEM2,ITEM3,ITEM4];LABEL#1;LABEL#2;(up to LABEL#9) RANDOMIZED_MULTISCALE;AFFECT SCALE;Please report your affect levels;[Valence,Arousal,Approach-avoidance motivation]Very low;Low;Medium;High;Very high	The same as MULTISCALE but the listed items are randomized upon each application start.	

INPUTFIELD INPUTFIELD;HEADER_TEXT;BODY_TEXT;BUTTON_TEXT;INPUTFIELD1_TEXT;INPUTFIELD2_TEXT INPUTFIELD;Study Data;Please enter data for this session;START THE STUDY;ResearcherID;ParticipantID;Session Nr;Additional comments	Displays a header, text, up to 10 input fields, and a button with text. The user taps the field and inputs text data. This can be used by the participant (e.g., to provide qualitative data) or by the researcher (e.g., to enter a participant's ID).	
Metadata		
LOG LOG;Text LOG;The first block starts here	Logs any predefined text in the output and when this part of the protocol was reached. It can be used to increase the readability of the output.	
STUDY_ID STUDY_ID;Text STUDY_ID;MyFirstOutdoorStudy_labphone1	Optional. If used, each generated output file will start with this text. It helps to organize the collected data, mainly if more labphones are used (e.g., social experiments)	
// This is ignored --- This is also ignored This is also ignored because anything that does not start with a command is ignored.	Lines that do not start with any command will be ignored. Thus, any characters can be used for commenting and syntax readability in line with personal preferences.	
END	Optional. Commands placed after this command will not be executed. It is helpful when the user wants to retain some commands in the bottom part of the protocol file, e.g., for later use during protocol development or testing.	

HelA (Helpful Assistant for PoLA) Documentation

Introduction

HelA, or Helpful Assistant, is an AI-powered support tool designed to assist researchers in creating and revising experimental protocols for use with the Pocket Lab App (PoLA). HelA simplifies the transition from research ideas to actionable experimental protocols by leveraging natural language processing and understanding. This documentation outlines HelA's capabilities, usage, and how it integrates with PoLA to facilitate efficient experimental setup and execution.

Features

- **Natural Language Understanding:** HelA interprets instructions provided in natural language, allowing researchers to describe their study requirements without writing in code or specific command syntax.
- **Protocol Generation**: Automatically generates PoLA-compatible protocols based on descriptions of the study design, objectives, and required measurements.
- **Protocol Revision:** Offers suggestions for improving existing protocols by identifying potential issues or opportunities for refinement.
- **Questionnaire Conversion:** Converts traditional questionnaire formats into interactive PoLA slides, selecting suitable command formats based on the content.
- **Creative Solution Finding**: Suggests workarounds for functionalities not directly supported by PoLA, enhancing the versatility of experiment design.

Getting Started with HelA

To utilize HelA, researchers must have a basic understanding of their research objectives and the types of data they wish to collect. HelA operates through a user-friendly interface where researchers input descriptions of their study in natural language or paste their protocols for reviewing.

Accessing HelA

HelA can be accessed as a chatGPT tailored GPT (via the GPTs functionality). Visit PoLA website for the link or use this link: https://chat.openai.com/g/g-Vz0JnWtqf-pola-helpful-assistant-hela
You need to have access to ChatGPT premium account to use GPTs. Upon accessing HelA, you will be greeted with an input field where you can start describing your experimental needs.

Inputting Instructions

1. (optional) Describe Your Study: Include a brief overview of your study, including its purpose and the data you intend to collect.
2. Detail Specific Requirements: Describe point-by-point specific elements of your study design, such as the types of questions you plan to ask, the scales to be used for responses, timings for timed tasks, and any special instructions for participants.
3. Specify Desired PoLA Commands: If you're familiar with PoLA's capabilities, you can mention specific commands you wish to use. However, this is unnecessary, as HelA can suggest the most appropriate commands based on your description.

Receiving and Implementing Suggestions

After inputting your instructions, HelA processes the information and generates a suggested protocol. This protocol will be presented in a format compatible with PoLA, including all necessary commands and settings. Review the suggestions carefully and adjust as needed to fit

your research design. You can also paste the protocol to another instance of HelA for a second critical review.

Creativity and Troubleshooting

HelA is capable of creative problem-solving, offering novel solutions to research design challenges. It can suggest alternative approaches when specific experimental designs cannot be directly implemented in PoLA. However, some of the creative solutions should be inspected critically.

Best Practices for Using HelA

- **Be Specific:** Provide detailed descriptions of your study requirements to enable HelA to generate the most accurate and valuable protocols.
- **Be systematic:** Split your instruction into sections that follow specific slides on the smartphone.
- **Iterate:** Generate a protocol, review it, and refine your description based on the output. Iterative design helps fine-tune the protocol to your precise needs.
- **Verify:** Always verify the generated protocol manually to ensure it meets your study's objectives and aligns with ethical guidelines.
- **Copywrite:** Do not paste copyrighted material for LLMs that use users' input for training.

Limitations

While HelA significantly streamlines the process of protocol generation, it is essential to recognize its limitations. HelA's suggestions are based on its current understanding and the information provided by the user. Therefore, the accuracy and appropriateness of its recommendations depend heavily on the quality and detail of the input it receives.

Researchers are encouraged to use HelA as a complementary tool in their protocol development process, always applying their expertise and judgment to the final experimental design.

Conclusion

HelA offers a groundbreaking approach to experimental protocol development, making it easier for researchers to translate their ideas into actionable studies. By combining the power of AI with the simplicity and versatility of PoLA, HelA empowers researchers to conduct sophisticated experiments with minimal setup time and technical expertise required.

License and Disclaimer Addendum

GNU General Public License (GPL)

PoLA and HelA are distributed under the GNU General Public License (GPL) version 3. This licensing framework is dedicated to ensuring the freedom to share and change all versions of a program—to make sure it remains free software for all its users. Under the terms of the GPL:

- Freedom to run the program for any purpose.
- Freedom to study how the program works and change it to make it do what you wish.
- Freedom to redistribute copies.
- Freedom to distribute copies of your modified versions to others.

The source code for PoLA and HelA is available, allowing researchers and developers to inspect, modify, and improve the software according to their needs and share their modifications under the same GPL license.

For the full GNU General Public License, version 3, please refer to the GNU official website.

Disclaimer

PoLA and HelA are provided "AS IS" without warranty of any kind, express or implied, including but not limited to the warranties of merchantability, fitness for a particular purpose, and noninfringement. In no event shall the authors, copyright holders, developers, or any parties involved in the creation, production, or delivery of the software be liable for any claim, damages, or other liabilities, whether in an action of contract, tort, or otherwise, arising from, out of, or in connection with the software or the use or other dealings in the software.

Additional Notes

- **Use at Your Own Risk:** Users employ PoLA and HelA at their own risk. Users are responsible for the ethical and legal aspects of their research protocols. The developers strongly recommend reviewing generated protocols for compliance with ethical guidelines and research standards.
- **No Warranties:** The developers do not guarantee the software will meet users' requirements, operate under specific conditions, or operate uninterrupted or error-free.
- **Limitation of Liability**: To the fullest extent permitted by law, the developers shall not be liable for any direct, indirect, incidental, special, exemplary, or consequential damages (including, but not limited to, procurement of substitute goods or services; loss of use, data, or profits; or business interruption).

Contribution and Feedback

Users and developers are encouraged to contribute to PoLA's and HelA's development by providing feedback, bug reports, and code contributions. These contributions are invaluable to the software's continuous improvement and are welcomed under the same GNU GPL framework to ensure the software remains free and open for all users.

By respecting these guidelines and the GPL, we aim to foster a collaborative, open, and ethical environment for advancing scientific research and experimentation.