

Guide for Researchers: Setting Up ScreenLife Capture

Developed by students and researchers at the Singapore University of Technology and Design, the ScreenLife Capture application is an open-source software suite which allows researchers to collect screenome data from Android smartphones. Screenomes are sequenced high-frequency screenshots of participants' device use, and a technique to study digital media use pioneered by Reeves and colleagues (2021). For example, the ScreenLife Capture Android application captures a screenshot of a participant's smartphone use every five seconds. This allows for highly granular data on smartphone use which may be useful for researchers interested in the study of digital media.

ScreenLife Capture is the first freely available and open-source screenome collection tool. The application is free to download, edit, and use for academic research purposes. The full details of the application can be found in the following paper. Please use the following citation if you are using *ScreenLife Capture* for your research project:

Yee, A. Z. H., Yu, R., Lim, S. S., Lim, K. H., Dinh, T. T. A., Loh, L., Hadiananto, A., and Quizon, M. (2022, August 31). *ScreenLife Capture: An open-source and user-friendly framework for collecting screenome data from Android smartphones*.
<https://doi.org/10.31234/osf.io/sphq4>

If you want to use ScreenLife Capture for a research project, this document provides a simple step-by-step guide for you to set up your own app and connect it to your Google Cloud account. There are two things you will require before you can get started: (1) Google Cloud account and (2) Latest version of Android Studio.

Contents in this document:

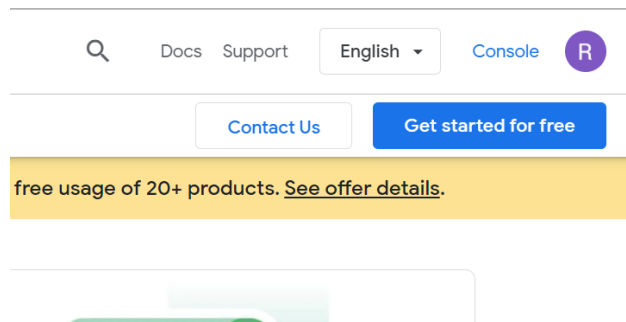
Step 1: Creating a Google Cloud Bucket.....	2
Step 2: Creating required service accounts and obtaining keys	5
Step 3: Setting up Cloud Functions	8
Step 4: Building the Android APK file for research participants.....	14
Step 5: Setting up the DMPO	15
Step 6: Registering a participant.....	16
Step 7: Downloading, decrypting, and obfuscating data	19

*****NOTICE***** We are looking for software engineers and data scientists to collaborate in order to further improve the framework (both software development and machine learning). We are also looking for PhD students to be part of the ScreenLife Capture team. Please email me at andrew.yee@sutd.edu.sg if you are interested to find out more.

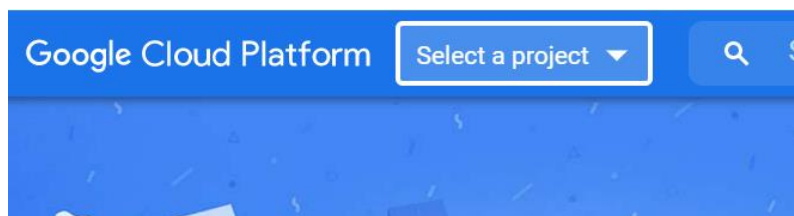
*****NOTICE*****

Step 1: Creating a Google Cloud Bucket

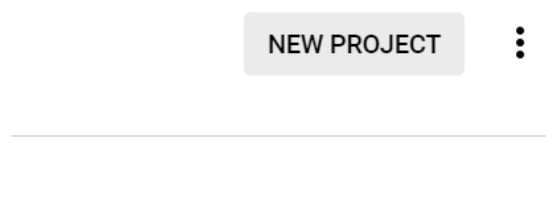
1. To begin, go to <https://cloud.google.com/> and either sign-in with your Google account or create a new account.
2. Once you are logged in, click “Console” on the top-right of the screen to enter your dashboard.



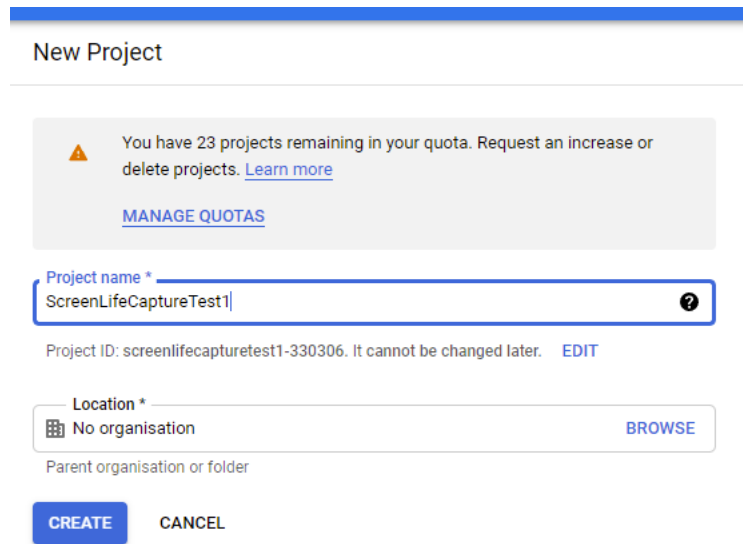
3. Click “Select a project” on the top-left of the screen.



4. To start a new project, select “New Project” on the top-right of the pop-up

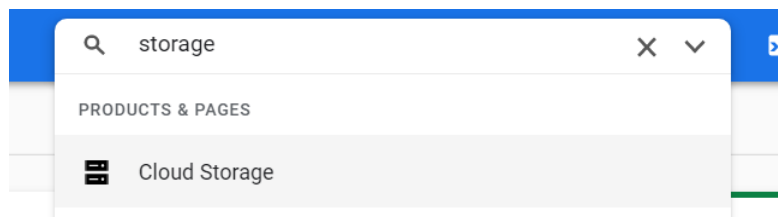


5. Fill in the “Project Name” and “Organization” as required and create a new project.

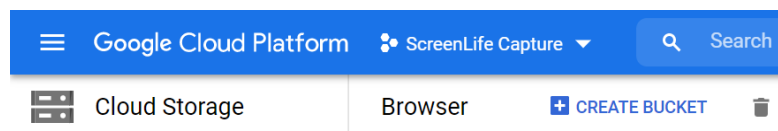


The screenshot shows the 'New Project' page in Google Cloud Platform. At the top, there's a blue header with 'New Project'. Below it, a warning message states: 'You have 23 projects remaining in your quota. Request an increase or delete projects. [Learn more](#)'. A link 'MANAGE QUOTAS' is also present. The 'Project name' field is filled with 'ScreenLifeCaptureTest1'. Below the name, it says 'Project ID: screenlifecapturetest1-330306. It cannot be changed later. [EDIT](#)'. The 'Location' dropdown is set to 'No organisation' with a 'BROWSE' button. At the bottom, there are 'CREATE' and 'CANCEL' buttons.

6. Go to “Cloud Storage” to create a cloud storage bucket for the project. You will require a billing account with Google Cloud at this stage. If you do not have a billing account, create one and link the project to the billing account before returning to this point of the setup.



7. Click “Create Bucket” at the top of the screen.



8. Choose a unique bucket name and select the “Region” location type.
9. Select the “location” closest to where the study will be conducted. Keep the “default storage class” as “Standard”.
10. Ensure “Enforce public access prevention on this bucket” is enabled, and that “Access control” is set to “Uniform”. “Choose how to protect object data” / “Advanced settings” can be ignored.
11. Click the “Create” button situated at the bottom left. Your cloud storage for this project has been created.

The screenshot displays the Google Cloud Storage 'Create bucket' wizard, which is divided into three main sections:

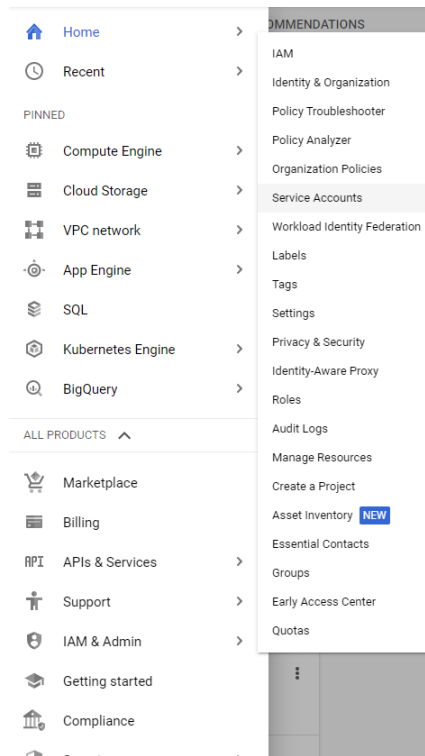
- Name your bucket:** This section prompts the user to enter a globally unique, permanent name. The example 'screenlifecapturetest1-bucket' is shown in a red-bordered input field, with a warning icon and text stating 'That bucket name is taken. Try another.' Below the input field, there is a 'CONTINUE' button.
- Choose where to store your data:** This section explains that the choice defines the geographic placement of data. It offers three 'Location type' options: 'Multi-region' (highest availability), 'Dual-region' (high availability and low latency), and 'Region' (lowest latency, which is selected). A 'Location' dropdown menu is set to 'asia-southeast1 (Singapore)'. A 'CONTINUE' button is at the bottom.
- Choose a default storage class for your data:** This section explains that the storage class sets costs for storage, retrieval, and operations. It lists four options: 'Standard' (selected, best for short-term storage), 'Nearline', 'Coldline', and 'Archive'. A 'CONTINUE' button is at the bottom.

Below the storage class section, there are two more sections:

- Choose how to control access to objects:** This section includes 'Prevent public access' (checked) and 'Access control' (set to 'Uniform'). A 'CONTINUE' button is at the bottom.
- Choose how to protect object data:** This section shows 'Protection tools: None' and 'Data encryption: Google-managed key'. At the bottom are 'CREATE' and 'CANCEL' buttons.

Step 2: Creating required service accounts and obtaining keys

1. From the menu on the left, select “IAM & Admin”, followed by “Service Accounts”.



2. Next, click the “Create Service Account” button near the top of the screen.



3. Enter a “service account name”. This account will be used to access the encrypted image files of the participants. Click “Create And Continue”.

1 Service account details

Service account name
for-admin

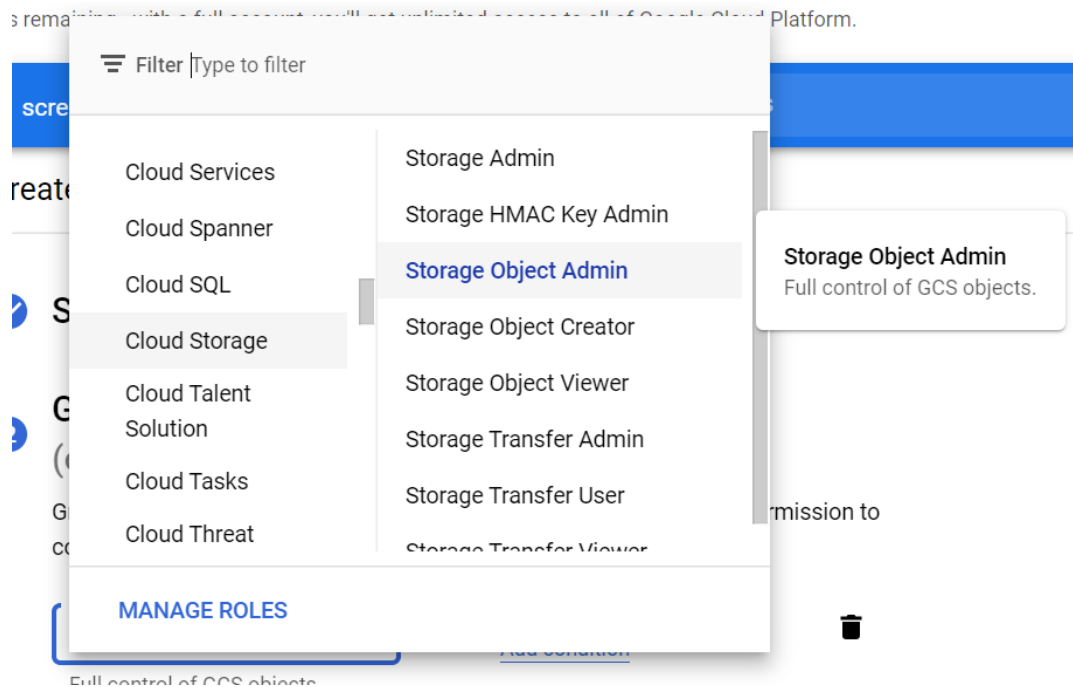
Display name for this service account

Service account ID
for-admin @screenlife-project.iam.gserviceaccount.com X ↺

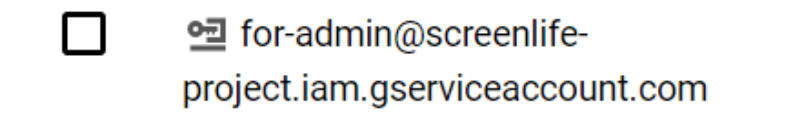
Service account description
Describe what this service account will do

CREATE AND CONTINUE

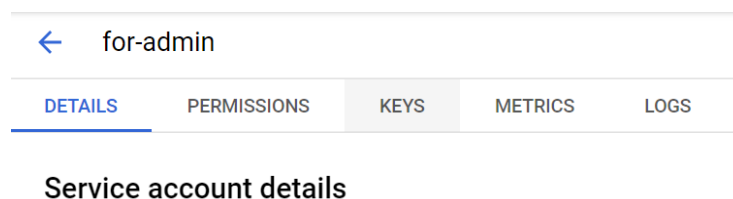
- Under “Select a Role”, select “Cloud Storage - Storage Object Admin”. Click “Continue”. Ignore “Grant users access to this service account”, then select “Done”.



- Click on the “email” section of the new account.

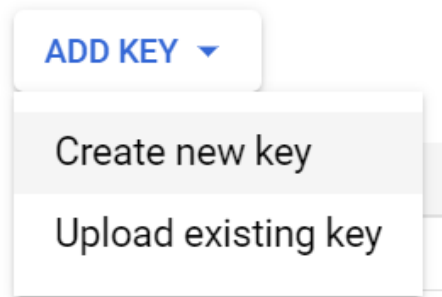


- Select the “Keys” section.

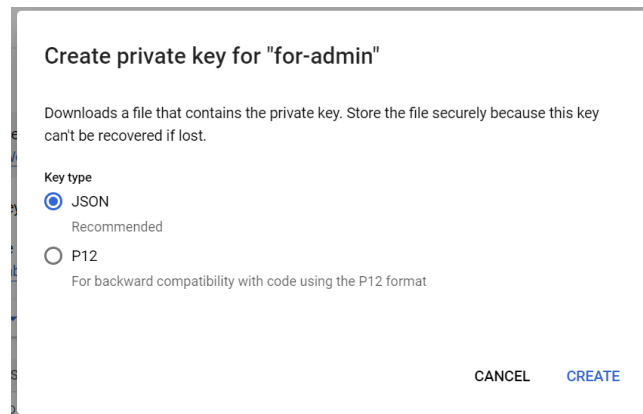


7. Select “Add Key” and then “Create Key”.

[Learn more about setting orga](#)



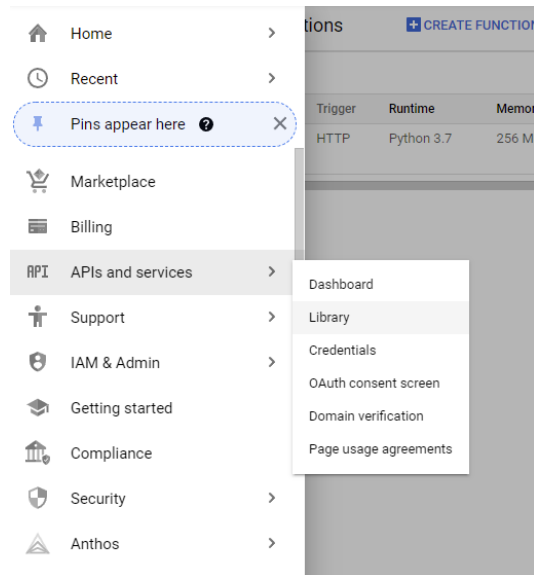
8. Ensure “JSON” is selected and click “Create”.



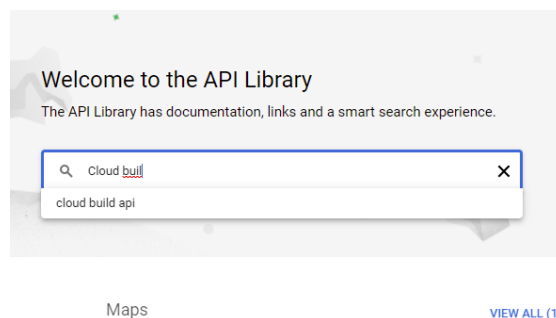
9. The downloaded .json file should be kept secure, as it will grant the ability to access the encrypted images stored in the storage bucket.

Step 3: Setting up Cloud Functions

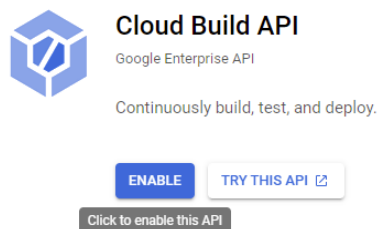
1. Return to the main page by clicking the “Google Cloud Platform” logo near the top-left of the screen. In menu on the left, under “APIs and services”, select “Library”.



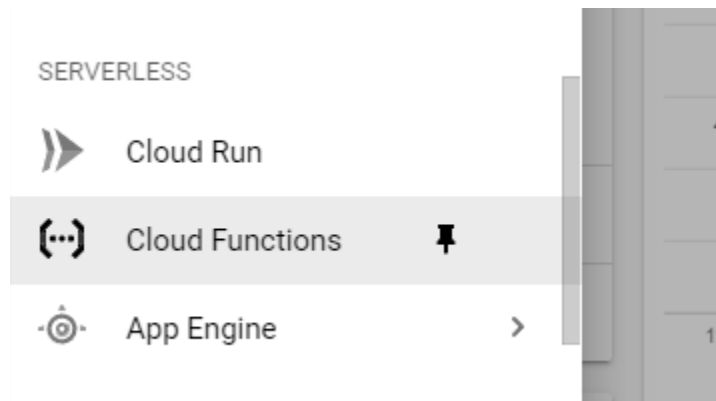
2. Search for “Cloud Build API” in the API library search bar.



3. Select “Cloud Build API” and enable the API for the project.



- Return to the main page by clicking the “Google Cloud Platform” logo near the top-left of the screen. In the menu tab, select “Cloud Functions”.



- Click the select “Create Function”. Under “Function name”, name it “upload”. Choose the region closest to the location of your study.

Basics

Function name *
upload

Region
asia-southeast1

- Under “Authentication”, check “Allow unauthenticated invocations. Check “Require HTTPS”, and click “Save”.

Trigger

⌕ HTTP

Trigger type
HTTP

URL

https://asia-southeast1-screenomics-sutd.cloudfunctions.net/hello-1

Authentication

☒ Allow unauthenticated invocations
Check this if you are creating a public API or website.

☐ Require authentication
Manage authorized users with Cloud IAM.

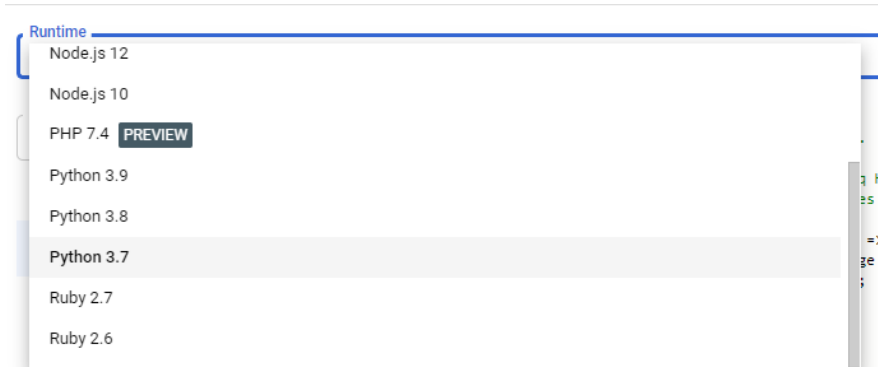
☒ Require HTTPS ?

SAVE CANCEL

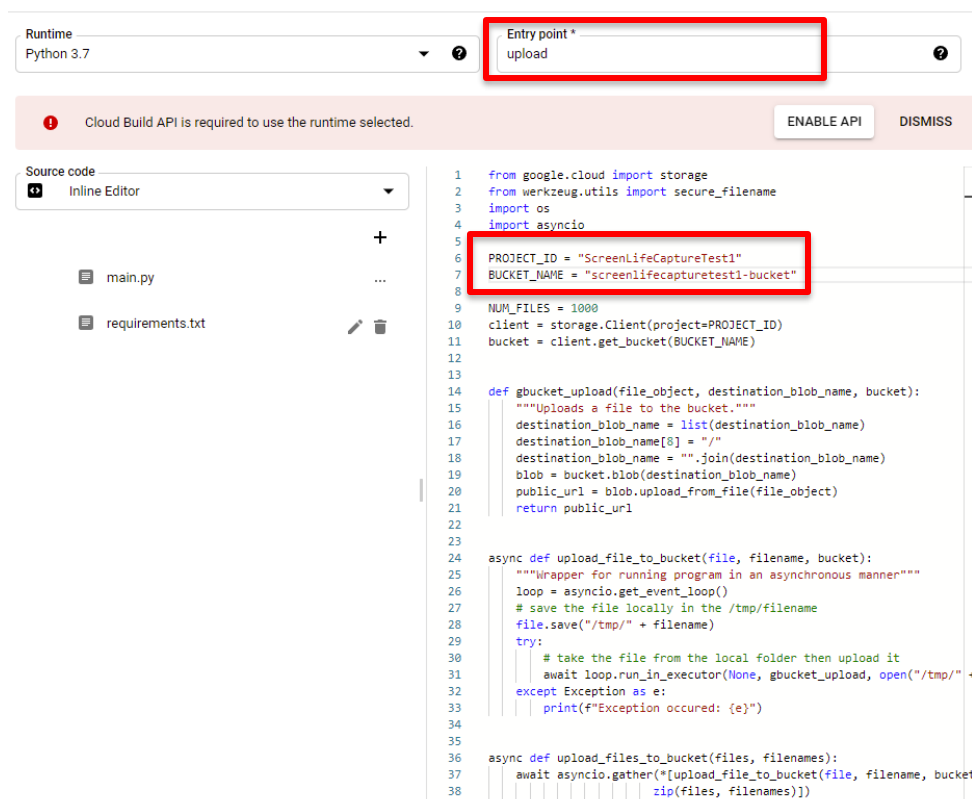
13. Click the “Next” button at the bottom-left of the screen.



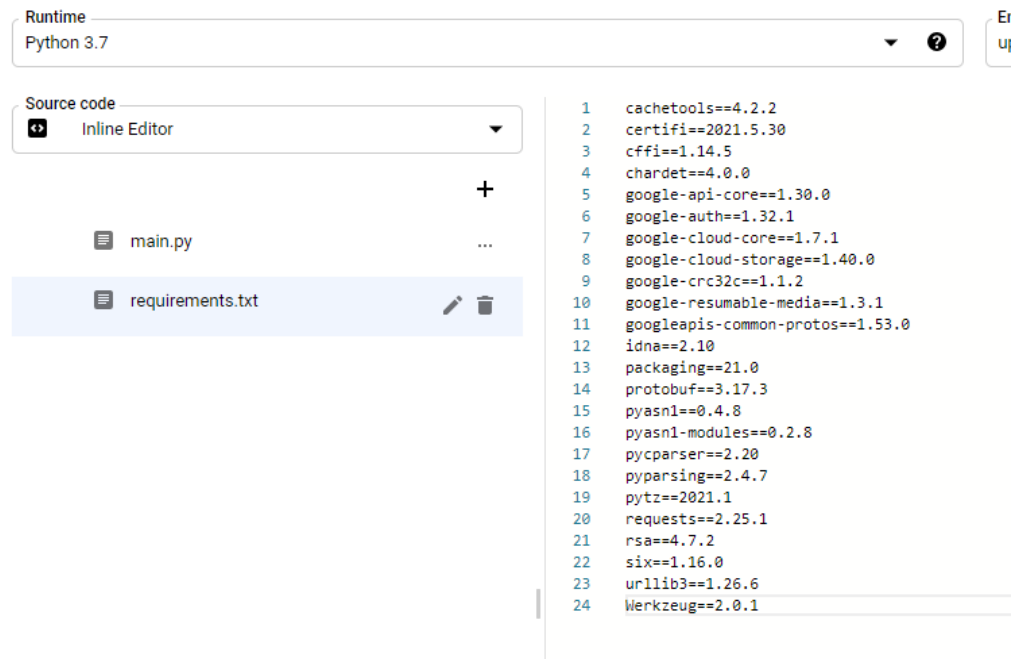
14. Under “Runtime”, select “Python 3.7”



15. Copy the code from “[main.py](#)” in the “[cloud-functions](#)” repository into the right-pane of the window, as shown in the image below. Modify the “BUCKET_ID” and “PROJECT_NAME” at the top of the pasted code to reflect the bucket ID and project name you had chosen previously. Then change the “Entry point” on the top-right of the screen to “upload”.



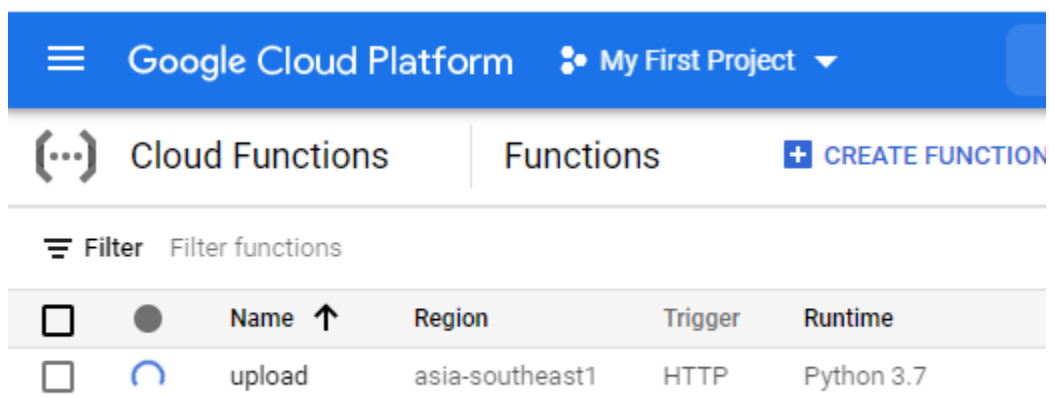
16. Click “requirements.txt” on the left side of the screen, and copy the code from “requirements.txt” in the “cloud-functions” repo onto the right-pane of the window, as shown in the image below.



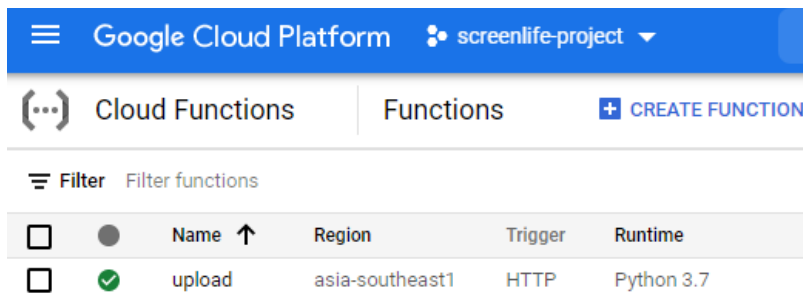
17. Click the “Deploy” button at the bottom of the screen.



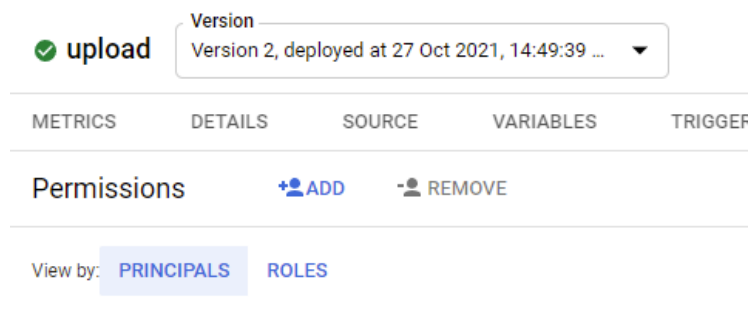
18. Wait for the function to start.



19. Click “upload”, the name of the function that was just created.



20. Under “Permissions”, click the “Add” button.



21. Fill in the “New principals” field with “allUsers”, and the “Role” field with “Cloud Function Invoker”. Click “Save”.

Add principals to upload

Add principals and roles for upload resource

Enter one or more principals below. Then select a role for these principals to grant them access to your resources. Multiple roles allowed. [Learn more](#)

New principals

allUsers

Role

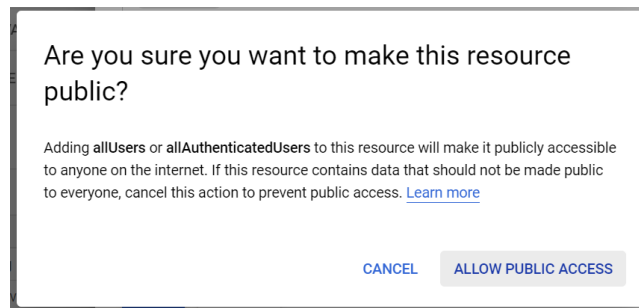
Cloud Functions Invoker

Ability to invoke HTTP functions with restricted access.

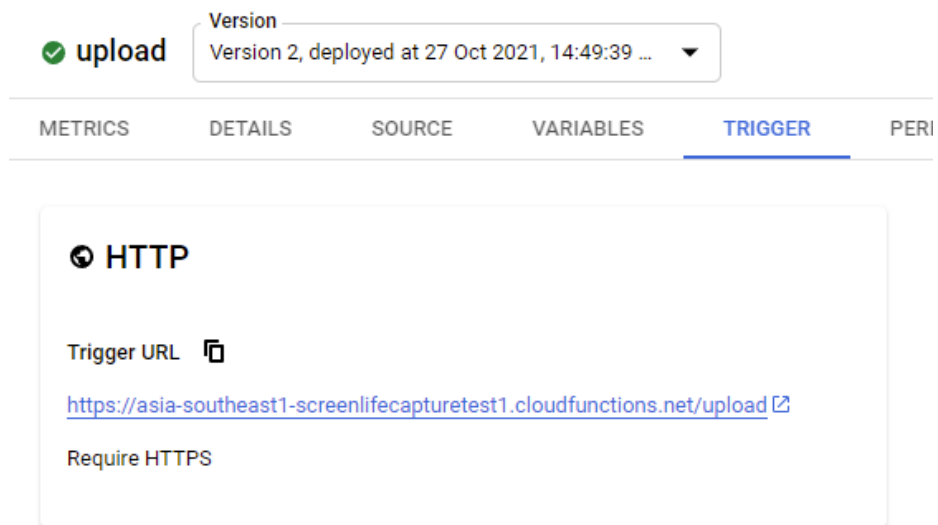
+ ADD ANOTHER ROLE

SAVE CANCEL

22. Click “Allow Public Access”.



23. Under “Trigger”, take note of the “Trigger URL”. This URL will be used when preparing the Android application for deployment.



Step 4: Building the Android APK file for research participants

1. Download and install [Android Studio](#). Ensure the software is up-to-date and set up using the standard settings.
2. Download the source code from the GitHub repository (link [here](#)) by clicking the green “Code” button, then clicking “Download ZIP”.
3. In Android Studio, open the “app-main” folder. Download and install dependencies if Android Studio prompts you. If prompted, update the Android Gradle plugin.
4. Navigate to “*android-app/app/src/main/java/com/screenomics/Constants.java*”. This file contains details which need to be updated to allow the app to access the cloud functions set-up previously. Insert the Trigger URL you obtained from the end of Step 3 into the `UPLOAD_ADDRESS` variable.
5. Make any desired changes to the Android application. Below are some examples of what you could change:
 - a. *src/main/res/layout/activity_main.xml* contains the UI of the main “screen capture” page.
 - b. *src/main/res/layout/register.xml* contains the UI of the initial “Scan QR” screen.
 - c. *src/main/java/com/screenomics/InfoDialog.java* contains the message displayed when selecting the “information” button on the main screen.
 - d. *src/main/java/com/screenomics/Constants.java* contains some values involved in the sending of encrypted image data, such as the maximum number of images to send in a batch, the number of batches to send, etc.
6. Save any desired changes.
7. Build the application.
 - a. Go to “Build”, “Build Bundle(s) / APK(s)”
 - b. Select “Build APK” and let Android Studio run.
8. Once the APK file is located, this should be sent to your participants after you receive their informed consent for them to install on their devices.

Step 5: Setting up the DMPO

1. Create a common folder to hold the DMPO software and the censoring script (e.g. "C:\Desktop\ScreenLife")
2. Download the DMPO software (.zip folder) from the [DMPO](#) repo and extract it into that common folder. You can also download the [censoring scripts](#) and extract it into the same common folder. Following that, there should be two subfolders in the "ScreenLife" folder ("DMPO" and "censoring-scripts").
3. Place the downloaded key file you obtained at the end of Step 2 into the "DMPO" folder. Rename the key file "bucket_key.json".
4. Install the latest [Java Development Kit \(JDK\)](#) and take note of its location on your computer.
5. In the "DMPO" folder, edit the "default-settings.json" file by filling in your Project ID, Bucket ID (which are your project name and bucket name in Google Cloud), and the location of the installed JDK under "javaPath" (it should look something like "C:/Desktop/JDK17/bin/java.exe").
6. Rename the "default-settings.json" file to "settings.json".

Step 6: Registering a participant

1. Start the DMPO software by following the instructions in the [DMPO](#) repo (start through Windows Powershell). Enter a 16-digit unique project passphrase when prompted. Keep this passphrase secure, as it will be used to access the management of the project.
2. Enter the passphrase on the input near the top-right of the window. Click “Submit Passphrase” when done. If not done before, this will be the passphrase for the rest of the project, and should be used when onboarding other participants, as well as during the decryption process.

Submit Passphrase

3. Click “Onboard Participant” near the top-right of the screen.

Onboard Participant

4. Fill in the participant ID and click “Register and Generate Keys”

Participant ID:

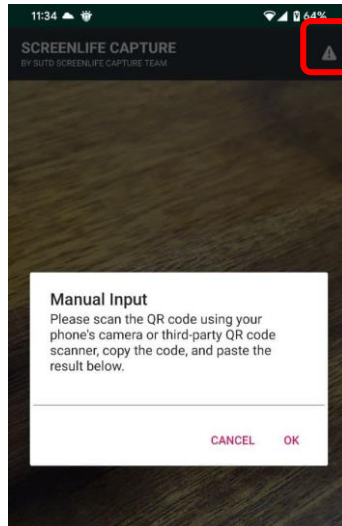
Register and Generate Keys

5. A QR code and verification code will be shown. After the participant receives and installs the “.apk” file, launch the application.
6. Let the participant scan the QR code shown in Manager. The button below should read “Verification code: XXXX”.

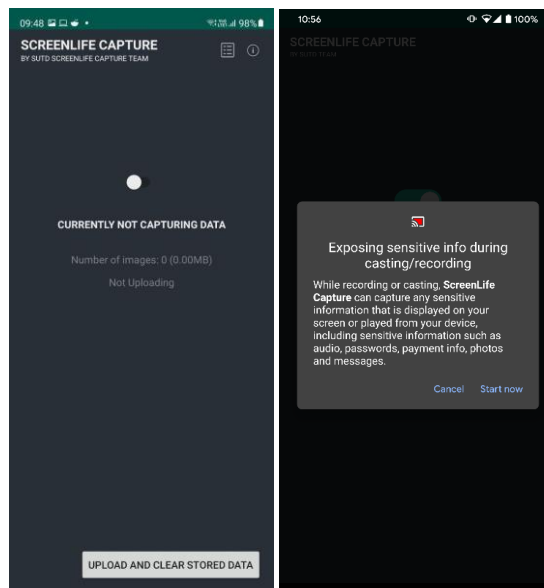


7. Confirm that the participant’s verification code is the same as that shown on the DMPO and prompt the participant to click the button to proceed.

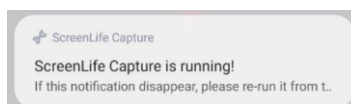
8. If the app cannot detect the QR code due to compatibility issues, users can use the device's camera or third-party/external QR code scanner to retrieve the unique key for manual input into the device by clicking the button on the top right corner.



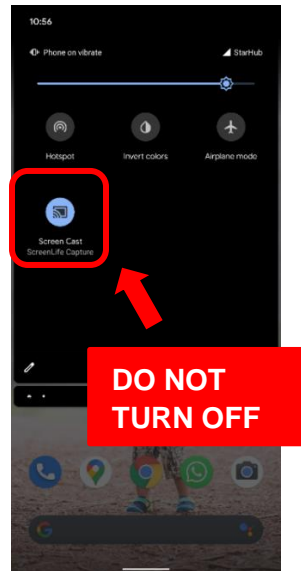
9. If the participant is onboarded, screen capture can then be enabled through the switch in the center of the screen. When prompted with the message that the smartphone will be exposing sensitive info, advise the participant to click "Start Now".



10. If successful, the participant should be able to see the following notification.



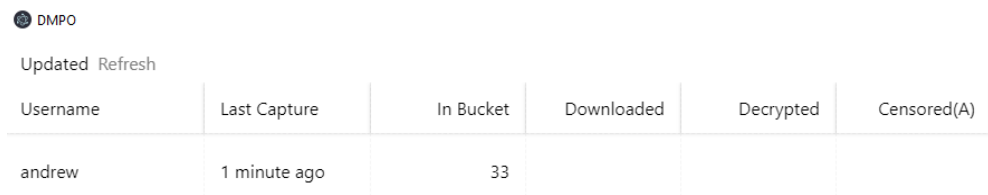
11. The screen cast function will automatically be switched on. Please advise the participant to keep it enabled throughout the entirety of the study. If it is turned off, the screenshots will not be collected.



12. Some smartphone models (e.g. Huawei, Xiaomi, Samsung, and Oppo) have proprietary battery optimization software which may kill the ScreenLife Capture app running in the background. It is advisable that participants using these smartphone models do two general things: (1) manually override the app settings and disable battery optimization for ScreenLife Capture, and (2) “lock” the app to prevent it from getting killed. The explanations and specific advice for different models of smartphones can be found on <https://dontkillmyapp.com/>.

Step 7: Downloading, decrypting, and obfuscating data

1. Launch the DMPO software.
2. Enter the passphrase on the input near the top-right of the window. Click “Submit Passphrase” when done.
3. You will be able to check the number of images in the storage bucket through the “In Bucket” column of the participant, as shown below.

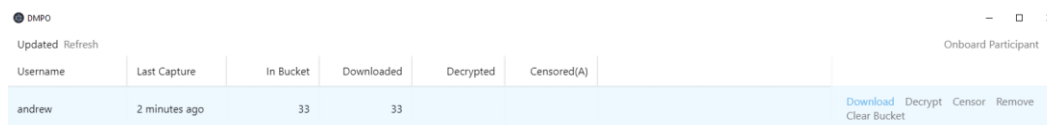


DMPO

Updated Refresh

Username	Last Capture	In Bucket	Downloaded	Decrypted	Censored(A)
andrew	1 minute ago	33			

4. To download the files for a single participant, click the “Download” button under the “Actions” column of the participant. (This button is hidden by default and will appear when hovered over.) The encrypted files will be in the “encrypted” folder.



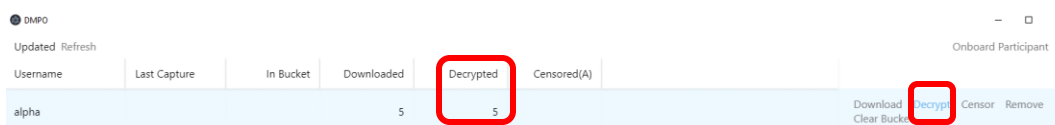
DMPO

Updated Refresh

Username	Last Capture	In Bucket	Downloaded	Decrypted	Censored(A)
andrew	2 minutes ago	33	33		

Download Decrypt Censor Remove
Clear Bucket

5. To decrypt the downloaded files, click the “Decrypt” button under the “Actions” column of the participant. The decrypted files will then be found in the “decrypted” folder.



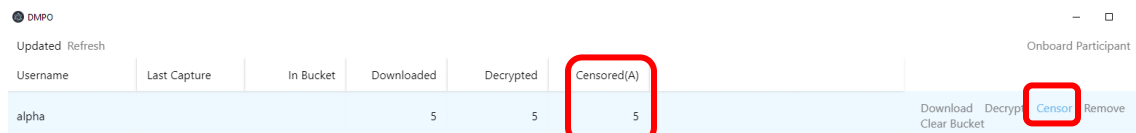
DMPO

Updated Refresh

Username	Last Capture	In Bucket	Downloaded	Decrypted	Censored(A)
alpha			5	5	

Download Decrypt Censor Remove
Clear Bucket

6. To use the automated censoring tool, follow the instructions here, then click the “Censor” button under the “Actions” column of the participant. The files with personal information removed will then be found in the “cleaned-automated” folder. **Please note that the automated personal information removal tool is still a work-in-progress. While usable, the accuracy rate is very low in our current testing. Please use this only for testing purposes.**



DMPO

Updated Refresh

Username	Last Capture	In Bucket	Downloaded	Decrypted	Censored(A)
alpha			5	5	5

Download Decrypt Censor Remove
Clear Bucket