

Android Data Recorder

Overview

This software is a data logger for android devices. It collects log data from one or more connected android devices and stores them into a database. Devices can either be connected via USB or network. For troubleshooting purposes, timestamps (markers) can be set, which allow the log data to be viewed in a self-defined period of time. The software also offers to record the screen of the connected devices and to create a video to a marker. Another function is the creation of issues. A backlog entry is created in JIRA, which contains selected markers, the associated log data for the selected time period, resources usage and, if desired, an MP4 video. So this software makes it easy for the user to find errors when testing android apps.

This software is designed to run on a Raspberry Pi or a computer of your choice.

Prerequisites

Download the latest [Android SDK Platform-Tools](#) and [FFMPEG](#) Version.

To start the software it is necessary to edit the config.json file which is located in the “/src” directory.

!! Please note necessary escape sequences !!

The config.json file has the following parameters:

Necessary:

Parameter Name	Description	Example
AdbPath	the path to the platform-tools adb executablefile	C:\\Program Files\\platform-tools\\adb.exe
FfmpegPath	the path to the Ffmpeg executable	C:\\Program Files\\ffmpeg\\ffmpeg.exe
VideoDirPath	the path where the video files are stored (directory seperator appended)	C:\\User\\Documents\\

Optional:

Parameter Name	Description	Default value
accessWorkloadInterval	Interval the CPU and Memory is refreshed (1-60s)	5
JiraServerUrl	URL to the JIRA Server	-
JiraUsername	JIRA Username (e-mail)	-
ApiToken	API Token from JIRA	-

How to build the software

```
git clone https://github.com/jsx15/AndroidDataRecorder.git
```

Switch directory to [...]AndroidDataRecorder/AndroidDataRecorder/

Build the software using dotnet:

```
dotnet build && dotnet run
```

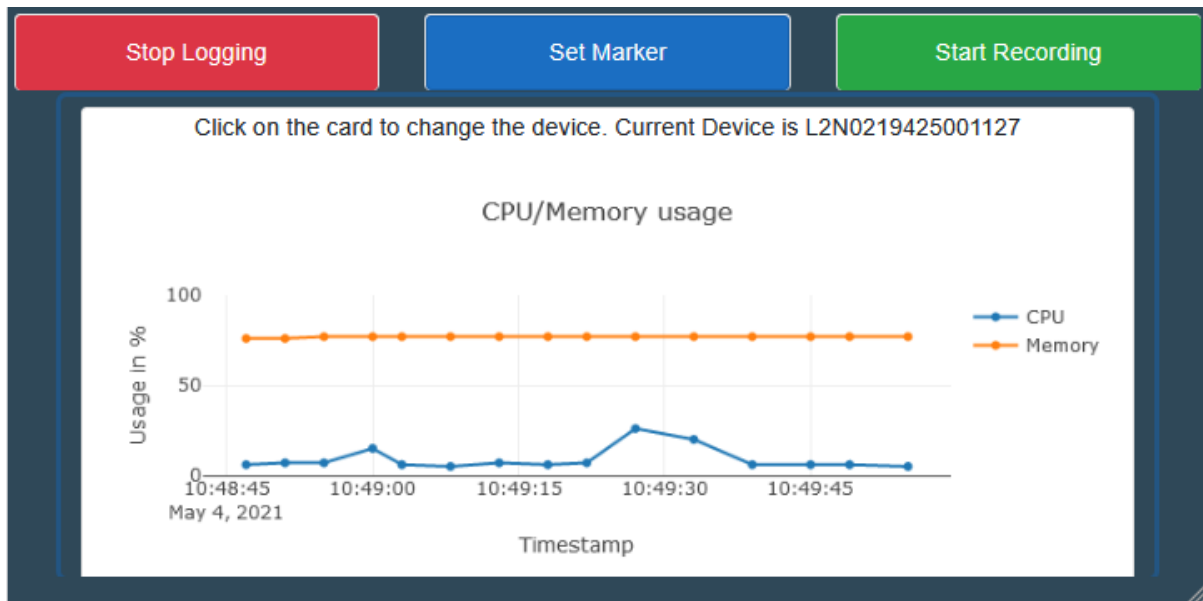
The software is accessible via <https://localhost:5001>.

General Instructions

Working on Raspberry Pi

The software can also be easily built and run on a Raspberry Pi.

If the Raspberry Pi has a screen attached there is a special page designed for this screen. The page can be accessed via <https://localhost:5001/pi>.



Note: The page is an absolute page. This means, when Stop/Start Logging or Start/Stop Recording is pressed logging or recording is started or stopped for every device attached. Markers are also set for every device attached.

The CPU and Memory usage for a current selected device can also be seen on that page. Change the device by clicking on the line chart.

Working on a Computer

Browse to <https://localhost:5001/> to open the web-app.

Dashboard

The upper-left side of the dashboard shows an overview about the currently connected devices. Each device connected to the host is listed here and logging is automatically started for each device.

- Logging can be stopped and started separately for every device by clicking the **"Start/Stop Logging" button**.
- Screen Recording can be stopped and started separately for every device by clicking the **"Start/Stop Recording" button**
- A device, that is connected via wifi, can be disconnected by clicking the **Disconnect** button.

To select a device, which is necessary for setting markers and displaying resources usage, a simple click on the device name is needed and the selection will be active. On the bottom of the card the selected device can be seen.

Overview

☒ HWMAR - L2N0219425001127

Stop Logging

Start Recording

Disconnect

Selected device is HWMAR - L2N0219425001127

In the upper-right side markers can be placed. This is achieved by simply entering a description and pressing the **Add Marker** button. The current time and date are shown below the input field. By clicking **Show All Markers** the user is forwarded to the Marker page (for further instructions see *Marker*).

Marker

03.05.2021 08:40:53

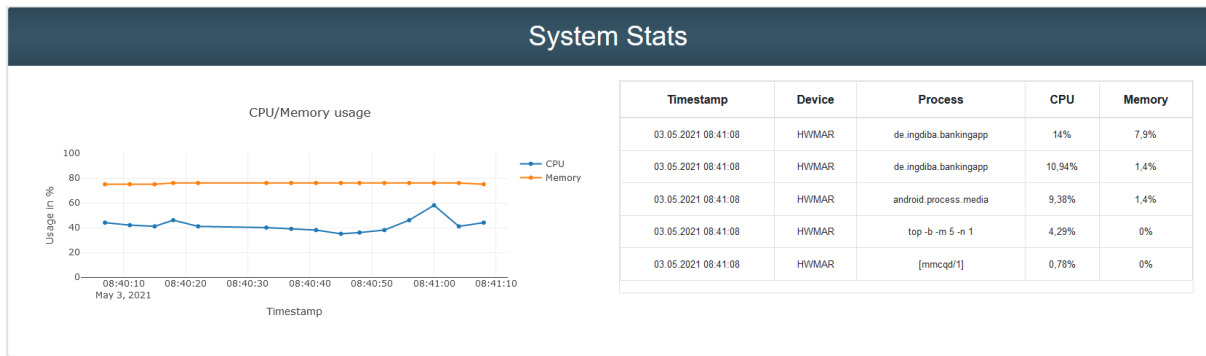
Add Marker

Show All Marker

The **System Stats** section shows a chart on the left and table with 5 rows on the right. The chart shows the CPU and Memory usage at a range of 0% - 100% of the selected device with a history of 15 timestamps. By hovering over the points the exact usage is shown.

The table displays the current 5 most expensive processes on the selected device with their cpu and memory usage. The range is the same as in the chart.

Both, the chart and the table are refreshed either configured in the config.json file or by default 5 seconds.



Marker

On the **Marker** page every marker is listed for the selected device. The device can be changed by clicking the dropdown in the upper-left.

As can be seen in the picture every marker appears with its ID, timestamp and message. By clicking the red cross at the right the marker is deleted. It is not possible to restore the deleted marker.

Markers can be selected by simply clicking on the row, which then turns blue as a hint for selection.

After finishing the selection of all markers needed for further steps, the **Apply** button is forwarding to the **Logger** page.

List				
Device: <div>HWMAR - L2N0219425001127</div>				
ID	Time	Message		
8	03.05.2021 08:43:08	Pressed Start Button		
9	03.05.2021 08:43:55	Clicked on epxport button after adding values		
10	03.05.2021 08:44:04	Marker from Pi		
11	03.05.2021 08:44:14	System Crashed		
<div>Apply</div>				

Logger

On this page, the log data associated with the markers can be filtered and displayed according to a time period and the log level. Each marker is handled individually, which makes it possible to get only the requested log data that is needed at that marker. It is also possible to use a marker twice with different settings.

All markers selected on the **Marker** page appear automatically on the **Logger** page. By clicking **Add Filter** more filters can be added.

By pressing **Apply filter**, the filter is applied and the associated log data is displayed in a table below the filter settings. **Add ticket** adds the marker with its filter selection to the ticket page (see *Ticket*). To switch to the Ticket page simply press **Go to ticket creation**.

The filtering system is quite easy:

- Marker: Shows the selected marker with its message
- Device: Shows the device associated to the marker
- Timespan - (Minutes): The time in minutes *before* the marker
- Timespan + (Minutes): The time in minutes *after* the marker
- Loglevel: The loglevel of the logs (V, D, I, W, E, F)

Note: When a loglevel is selected, the higher priority log levels are always included in the output. F has the highest priority, V the lowest. This means that if log level E is selected, all logs with log levels E and F are also outputted. To get all logs of all levels V must be selected.

An example can be seen in the picture below:

Our system crashed, a marker was set and selected. This marker now appears on the logger page. Since it is not 100% certain, when the crash happened, we now want to see the log data in the time span 0.5 minutes before and 0.5 minutes after. Since we are dealing with a complete crash of the application, we only want to see the log data of level E and F, so we select E as the log level. After applying the filter, all log data recorded at this time will appear in the table.

Filter

Marker:
03.05.2021 08:43:55 | Clicked on expg

Device:
HWMAR - L2N0219425001127

Timespan - (Minutes):
0.5

Timespan + (Minutes):
0.5

Loglevel:
E

Apply filter

Add to Ticket

Marker:
03.05.2021 08:44:14 | System Crashes

Device:
HWMAR - L2N0219425001127

Timespan - (Minutes):
0.5

Timespan + (Minutes):
1

Loglevel:
D

Apply filter

Add to Ticket

Add Filter

Go to ticket creation

Log Table of 03.05.2021 08:43:55

Device Name	System Timestamp	Device Timestamp	PID	TID	Loglevel	App	Message
HWMAR	03.05.2021 08:44:25	03.05.2021 08:39:36	839	22854	E	HIDMPL_CORE	opening file: No such file or directory
HWMAR	03.05.2021 08:44:24	03.05.2021 08:39:36	19540	32106	E	Media/Metadata/RetrieveJNI	getEmbeddedPicture: Call to getEmbeddedPicture failed.
HWMAR	03.05.2021 08:44:24	03.05.2021 08:39:36	808	9052	E	Metadata/RetrieveClient	failed to extract an album art
HWMAR	03.05.2021 08:44:21	03.05.2021 08:39:36	839	1812	E	VDEC	[event_process]: [3081] ProcessEventFromCmdQueue() is failed
HWMAR	03.05.2021 08:44:21	03.05.2021 08:39:36	839	1812	E	VDEC	[event_process]: [3081] ProcessEventFromCmdQueue() is failed
HWMAR	03.05.2021 08:44:19	03.05.2021 08:39:36	1224	1720	E	WifiBandControl	Notice: 0, Smr: 0, Chloaid: -1
HWMAR	03.05.2021 08:44:04	03.05.2021 08:39:36	839	839	E	libc	Access denied finding property "vendor.fold_displayflag"
HWMAR	03.05.2021 08:44:02	03.05.2021 08:39:36	839	839	E	HIDMPL_CORE	opening file: No such file or directory
HWMAR	03.05.2021 08:44:02	03.05.2021 08:39:36	19540	32106	E	Media/Metadata/RetrieveJNI	getEmbeddedPicture: Call to getEmbeddedPicture failed.
UABABD	03.05.2021 08:44:03	03.05.2021 08:39:36	808	91207	E	Metadata/RetrieveClient	failed to extract an album art

Ticket

This page is used to create a bug ticket from the previously filtered log data. It is possible to create a JIRA backlog entry. (see Prerequisites)

All filtered markers with their log data previously added to a ticket appear in the table on the page. Per marker the timestamp, message, number of logs and timespan are displayed. It is also possible to include the video file belonging to the marker. This is achieved by checking **Include Video**.

The name as well as the description of the entry can be entered in the input field provided for this purpose below the list. The priority of the entry can be selected from a dropdown as well as the type of the task. The project key, means the project for which the ticket is intended, can also be selected from a drop-down list (**Note:** The project keys, the task type and the priority levels are retrieved from JIRA and displayed there).

What happens when a ticket is created?

All log data and resource usage data associated with the markers are attached to the JIRA Backlog entry as a .txt or .json file. One log file is created and attached per marker. This means that if 3 markers are attached, the backlog entry will contain 3 log files. If **Include Video** is selected, a .mp4 file will also be attached. The text file is named after the device and the unique marker ID. The video file is named after the marker id and the device name.

The ticket .txt or .json file can be found in the Documents directory. The video is saved in your configured video directory.

JIRA Backlog Entry			
Marker 03.05.2021 08:43:55	Log Entries: 23	Create Video: <input checked="" type="checkbox"/>	Timespan: 1 minutes
Marker 03.05.2021 08:44:14	Log Entries: 738	Create Video: <input type="checkbox"/>	Timespan: 1.5 minutes

Entry Title:

Entry Description:

Issue Type: Priority:

Project Key: File Format:

How is the .txt file structured?

All information like device, time span, marker ID and marker message are in the header of the file. Between the log data the maker is inserted at its set place.

To find the marker despite a large number of log data, it is marked with

###jiraprojectkey###

This makes it much easier to find the slot the marker was set.

Resource usage can be find with

##jiraprojectkey##

How is the .json file structured?

Each object contains 13 values:

Not nullable:

Timestamp	Host time of creation
DeviceName	Name of the device
DeviceSerial	Serial number of the device

Nullable:

Message	Message of the log
DeviceTimestamp	Device time of creation
Pid	Process ID
Tid	Thread ID
LogLevel	Level of Log
App	App that caused the log
Cpu	CPU usage of the android device
Memory	Memory usage of the android device
Battery	Battery status of the android device
MarkerId	Id of the marker

Config

To connect a device via network it is required to add the device to the known devices list. Therefore the IP-address of the device is needed. The IP-address can be entered in the input field and the device is added by pressing **Add Device**.

Add Device

IP-address:

Add Device

Known Devices

123.123.123.123	<button>Attach</button> <button>Remove</button>
-----------------	---

ADB Version 1.0.41

In the Danger Zone, the ADB Server can be killed and restarted if any errors occur. Also the database can be cleared (all data will be lost. Forever!).

DANGER ZONE

Restart ADB serverKill ADB serverClear database