**Lab 18. Performance Analysis**

V1.0

30/05/2020

We finally have an application with UI, local DB, and network connection. It is the time to run the App and analyse its performance. The following indicators are considered.

* CPU
* Memory
* Network
* Energy

The Android Studio embedded tool “Profile” is used in this tutorial to collect data. Below are the topics of this tutorial.

**Topics**

[**1.** **Collecting performance data with “Profile” tool** 1](#_Toc41766075)

[**Step 1.** **Setup application.** 1](#_Toc41766076)

[**Step 2.** **Start “Profile” tool and collect data** 1](#_Toc41766077)

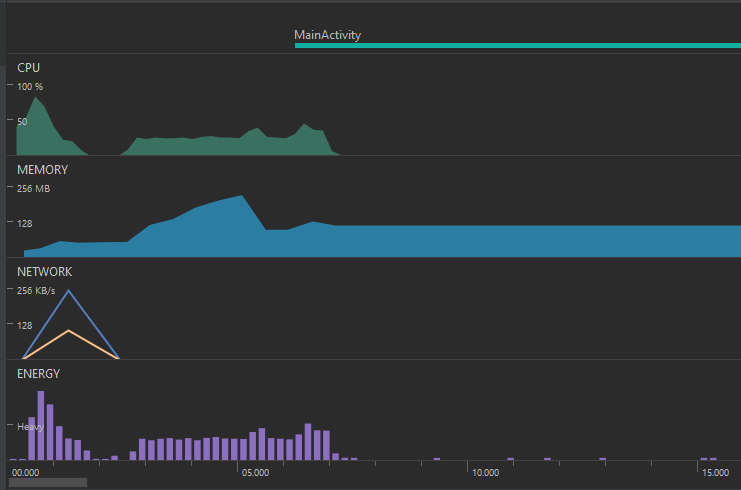
[**Step 3.** **Analysis the data** 2](#_Toc41766078)

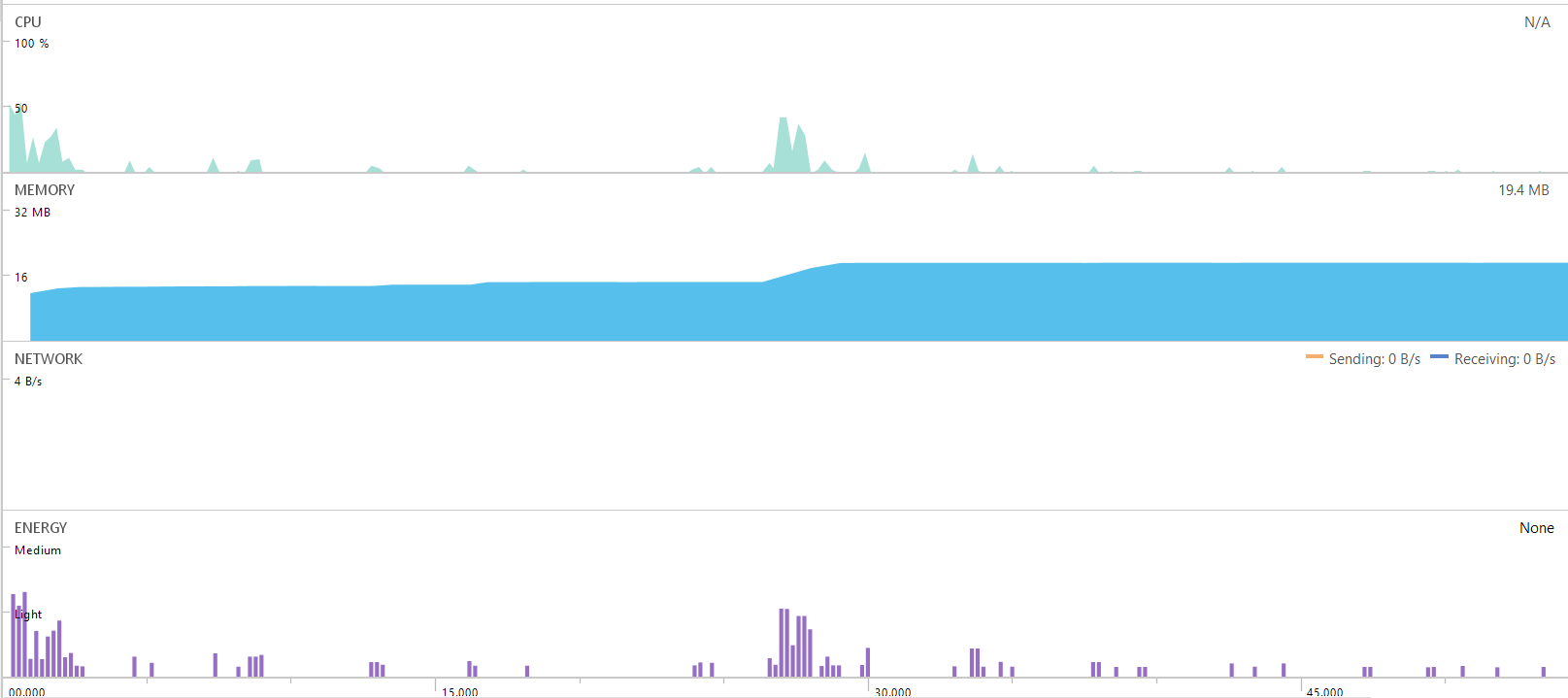
1. **Collecting performance data with “Profile” tool**
   1. **Setup application.**

It is recommended to launch an application with network interaction. So that download / upload data will be collected

* 1. **Start “Profile” tool and collect data**

Use this [tutorial](https://developer.android.com/studio/profile/android-profiler) to learn how to start the tool and how to collect data with it. The following data is collected for the start-up stage of the App created in Lab 17. In the App, CRUD function were performed when the MainActivity is launched.





* 1. **Analysis the data**

Here is an example how to analyse the data. First, identify what the action was taken. Then analysis the graph seen in the tool

***Action #1****: Start up the App*

**CPU:**

* During Start Up, CPU usage spike occurs at the 0.6 seconds with a peak value of 55%
* After 2.70s the CPU usage drops to 0%
* At approx. 4.00s, 8.00s and 12.00s there are some small CPU Usage instances (10% or less)
* Another CPU usage spike occurs at 27.00s with a peak usage of 43%
* CPU Usage stabilizes at 0% approx. after 50s

**Memory**

* Memory usage from 0.00s ~ 26.00s remains fairly constants around 14MB
* After 26.00s~29.00s, memory usage increases to 19.4MB and remains at 19.4MB for the remaining duration of the test.

**Network**

**Energy**

* Energy graph follows the same pattern at the CPU graph.

End of this tutorial