

Espen-Andreas Oseth,
Lars Martin Nygaard,
Vebjørn Albinson,
Wisam Khalid Razi

Monitoring of 4G connectivity in industrial use-cases

Bachelor's thesis in Digital Infrastructure and Cyber Security
Supervisor: Ernst Gunnar Gran
Co-supervisor: Henning Elvestad, Bjørn Isene and Andreas Rømo
May 2023

Espen-Andreas Oseth,
Lars Martin Nygaard,
Vebjørn Albinson,
Wisam Khalid Razi

Monitoring of 4G connectivity in industrial use-cases

Bachelor's thesis in Digital Infrastructure and Cyber Security
Supervisor: Ernst Gunnar Gran
Co-supervisor: Henning Elvestad, Bjørn Isene and Andreas Rømo
May 2023

Norwegian University of Science and Technology
Faculty of Information Technology and Electrical Engineering
Dept. of Information Security and Communication Technology



Abstract

Telenor is the biggest telecommunication operator in Norway, where they have built out a nationwide high-speed 4G network of world-class quality. 4G is used for many different purposes, one of which is operating automated vehicles in the context of industrial use-cases. While nationwide 4G coverage does not necessarily guarantee robust and high-quality network connectivity for devices connected to the 4G network, there are many factors that can affect the 4G signal, potentially leading to a lack of desired network quality. This is especially critical for automated vehicles, where a continuous high-quality 4G connection is crucial to perform their tasks effectively.

The purpose of this bachelor's thesis is to analyze how the quality of the 4G connection is affected by different factors that can interfere with the 4G signal. Various tests were performed to measure signal quality in Telenor's 4G network, aiming to imitate the conditions of automated vehicles in an industrial use-case, specifically in quarry areas. The test results served as a basis for discussing a "superparameter". The superparameter is intended to serve as a warning to the vehicle operators when the quality of the 4G connection is insufficient to operate the automated vehicles. The gathered results show that the parameters used to create the superparameter occasionally vary significantly in relation to each other, making it challenging to establish a single superparameter to determine 4G quality. Therefore, it is more reliable to monitor each parameter that affects 4G quality individually.

Sammendrag

Telenor er den største telekommunikasjons operatøren i Norge, hvor de har bygd et landsdekkende høyhastighet 4G-nettverk i verdensklasse. 4G-nettverket blir brukt til mange forskjellige formål, hvor et av disse formålene er å kunne bruke mobilnettverket til å drifte automatiserte kjøretøy i sammenhengen med industrielle bruksområder. Med landsdekkende 4G dekning betyr dette nødvendigvis ikke at enheter koblet til 4G-nettverket vil ha en robust og høykvalitet nettverkskonnektivet. Det oppstår ulike faktorer som kan påvirke kvaliteten på 4G signalet, som kan medføre at enheter ikke oppnår den nettverkskvaliteten de ønsker. Spesielt for automatiserte kjøretøy er det ekstra viktig å ha en kontinuerlig 4G forbindelse for å kunne holde driften i gang i disse industrielle bruksområdene.

Formålet med denne bachelor oppgaven er å analysere hvordan kvaliteten på 4G-nettverket blir påvirket av ulike forhold som kan påvirke radio signalet. Det ble utført ulike tester som målte signal kvaliteten i Telenors 4G nettverk, hvor det i løpet av testene ble forsøkt å etterligne forholdene man finner i industrielle bruksområder, mer spesifikt forholdene med automatiserte kjøretøy som blir brukt i forbindelse med et steinbrudd. Resultatene fra testen ble også brukt som et utgangspunkt for å kunne drøfte fra en «superparameter». Superparameteren skal fungere som et forvarsel til kjøretøys operatorene om når kvaliteten på 4G tilkoblingen ikke er god nok for å drifte kjøretøyene. Resultatene viser at parameterene som kan brukes til en superparameter vil tidvis variere for mye i forhold til hverandre til at én enkel superparameter alltid vil kunne gi et riktig svar på hvordan 4G kvaliteten er. Derfor vil det være bedre å overvåke hver enkel parameter individuelt.

Preface

The team would like to express their gratitude to the individuals who contributed to making this bachelor thesis possible:

Supervisor: NTNU - Ernst Gunnar Gran

We would like to thank Ernst for the continuous supervision and guidance throughout the project period. Without him, the team would not have been able to complete the project successfully.

Telenor - Henning Elvestad, Bjørn Isene, Andreas Rømo

We are grateful to Henning Elvestad, Bjørn Isene, and Andreas Rømo from Telenor for their expertise in Telenor's cellular networks. They provided invaluable resources and guidance throughout the project. Since the team had no prior knowledge of cellular networks, their expertise was crucial for the successful execution of the project.

Contents

Abstract	iii
Sammendrag	v
Preface	vii
Contents	ix
Figures	xiii
Tables	xvii
Code Listings	xix
Acronyms	xxi
Glossary	xxv
1 Introduction	1
1.1 Background	1
1.2 Assignment	2
1.2.1 Objectives	2
1.3 Target Audience	2
1.4 Our starting point	2
1.5 Roles outside the team	3
1.6 Scope	3
1.7 Thesis Structure	4
1.8 Notes	4
2 Theory	5
2.1 4G	5
2.1.1 4G Long Term Evolution	5
2.1.2 Base stations in 4G LTE	6
2.1.3 Radio Waves	6
2.1.4 Interference	7
2.1.5 Network congestion	7
2.1.6 Frequency Bands	7
2.2 3GPP frequency bands	9
2.2.1 Duplexing	9
2.2.2 800 MHz frequency band (B20)	10
2.2.3 1800 MHz frequency band (B3)	11
2.3 4G signal level measurement	11
2.3.1 RSRP - Reference Signal Received Power	11
2.3.2 RSRQ - Reference Signal Received Quality	12

2.4	Handover	12
2.4.1	Hysteresis margin	12
2.4.2	Time-to-trigger	13
2.4.3	Handover interruption time	14
2.5	Antennas	14
2.5.1	Antenna radiation pattern	14
2.5.2	Gain and types of antennas	15
2.6	Carrier aggregation	17
2.7	Cellular gateway	17
3	Methods	19
3.1	Hardware	20
3.1.1	Cisco Meraki cellular gateway (MG21)	20
3.1.2	Nordic Thingy 91	20
3.1.3	Smartphones	20
3.1.4	Laptops	21
3.1.5	Remote Server	21
3.2	Software and tools	21
3.2.1	Cisco Meraki cloud	21
3.2.2	Cisco Meraki RESTful API	21
3.2.3	Speedtest.net	22
3.2.4	iPerf3	22
3.2.5	GPSLogger	22
3.2.6	G-MoN Pro	22
3.2.7	Microsoft Teams	22
3.2.8	PuTTY	22
3.2.9	Elastic Stack	22
3.2.10	Python	23
3.2.11	OBS Studio	23
3.3	MG21 frequencies specification and radiation pattern	23
3.3.1	MG21 frequencies	23
3.3.2	internal Antennas and radiation pattern	23
3.3.3	RSRP and RSRQ values	25
3.4	Test area and signal readings from Telenor	25
3.4.1	Test Area	26
3.4.2	Best coverage eNodeB	27
3.4.3	800 MHz frequency band (B20) coverage	27
3.4.4	1800 MHz frequency band (B3) coverage	28
3.4.5	RSRP readings	29
3.5	Setup	30
3.5.1	Initially planned setup	31
3.5.2	Changes to setup	32
3.5.3	Final setup	34
3.6	Test execution	37
3.7	Data processing	37

3.8	The playbook	39
3.9	Hypotheses	39
3.9.1	Result compared with Telenor's previous TEMS data	39
3.9.2	Cell selection and handover	40
3.9.3	Driving direction	40
3.9.4	Bandwidth	41
3.9.5	RSRP and RSRQ data	42
3.9.6	Weather	43
3.10	Method summary	43
3.10.1	Final setup	43
3.10.2	Test execution	44
3.10.3	Data processing	45
3.10.4	Hypotheses	45
4	Results and Discussion	47
4.1	Showcasing overall result	47
4.2	Frequency Bands	50
4.2.1	Frequency bands utilized	50
4.2.2	Frequency band patterns	51
4.2.3	Occurrence of the 900 MHz frequency band	55
4.3	RSRP and RSRQ	58
4.3.1	MG21 with stronger RSRP	58
4.3.2	Varying values within geographical points	61
4.3.3	Weather	62
4.3.4	RSRQ inconsistency	62
4.3.5	Difference between 800 MHz- and 1800 MHz frequency bands	63
4.4	Handover and switches between frequency bands	72
4.4.1	New eNodeB's discovered	74
4.4.2	Bandwidth during handover	76
4.5	Upload speed	77
4.5.1	Frequency band and Bandwidth	78
4.6	Download speed	80
4.6.1	Frequency band and Bandwidth	81
4.7	Results summary	82
5	Superparameter	85
5.1	The different parameters	85
5.1.1	RSRP and RSRQ	86
5.1.2	Bandwidth	86
5.1.3	Frequency band	87
5.2	Why not just use bandwidth?	87
5.3	Proposition A	88
5.3.1	The formula	88
5.3.2	Testing the formula	89
5.4	Proposition B	91
5.4.1	Base variant	91

5.4.2	Variant with bandwidth	92
5.4.3	Testing the first and second variants	93
5.4.4	Variant with separate weighting	95
5.5	Proposition A vs Proposition B	97
6	Conclusion	99
6.1	Objectives	99
6.2	Testing and heatmaps	99
6.2.1	Choice of test area	99
6.2.2	Lack of data	100
6.2.3	Heatmap	100
6.3	The superparameter	100
6.4	Suitability of the MG21	101
6.5	Further work	102
	Bibliography	103
A	Assignment	109
B	Project Plan	115
C	Scripts and Logs	133
C.1	Test Script	133
C.2	Combine Script	140
C.3	Upload Script	143
D	Demo Setup	151
D.1	Setup Guide	151
D.2	Docker compose	157
D.3	Setup Script	159
E	Documentation	167
F	Playbook	175
F.1	Slides	175
F.2	Navigate Kibana	179
G	Timesheet	181
G.1	Overall	181
G.2	Espen-Andreas Oseth	187
G.3	Lars Martin Nygaard	192
G.4	Vebjørn Albinson	196
G.5	Wisam Khalid Razi	201
H	Meeting reports	207
H.1	Supervisor	207
H.2	Telenor	224

Figures

2.1	Example on what an eNodeB looks like	6
2.2	Illustration of the relation between high, mid, and low bands	8
2.3	Illustration of the handover process, HO=Handover, Hys=Hysteresis margin TTT=Time-to-trigger. Inspired by Telecommunication Systems [26, p.542].	13
2.4	XYZ-plane radiation illustration. Inspired by Wikipedia [30].	15
2.5	Antenna gain illustration. Inspired by Inhandnetworks [32]	16
3.1	Simplified figure of how the tests were performed.	19
3.2	Radiation pattern for the MG21 for the 829 MHz and 1940 MHz frequency ranges. Courtesy of Cisco Systems, Inc. Unauthorized use not permitted 15.05.2023 [64]	24
3.3	Test area used throughout all the tests. Taken from Google Maps	26
3.4	The placement of the eNodeBs relative to the test area. Skumsjovegen (A), Lauvkampen (B), Vardalsaasen (C). Taken from Google Maps	27
3.5	Illustration of which eNodeB is providing 800 MHz in each section of the test area. Lauvkampen (red), Vardalsaasen (orange), Skumsjovegen (yellow), taken from TEMS	28
3.6	Illustration shows the southernmost part of the test area (Figure 3.5) where there is 1800 MHz coverage from Skumsjovegen: the red area represents the 1800 MHz frequency band, and the orange area represents the 800 MHz frequency band. Taken from TEMS	29
3.7	RSRP values for the same section of the test area shown in Figure 3.6. The right star symbol represents RSRP for 1800 MHz, while the left star symbol represents RSRP for 800 MHz. Taken from TEMS	30
3.8	Color codes for TEMS reading related to figure 3.7	30
3.9	Illustration of the initial planned test setup.	31
3.10	How the hardware was initially set up in the car.	32
3.11	Comparison between the logged positions from the Thingy 91 and smartphone using GPSLogger	33
3.12	Technical outline of the final test setup.	35
3.13	The final hardware setup inside the car.	36
3.14	Illustration of the data processing process	38

3.15	Technical outline of the final test setup. (Reused Figure 3.12)	44
3.16	Simplified figure of how the tests were performed. (Reused Figure 3.1)	44
3.17	Illustration of the data processing process (Reused Figure 3.14) . .	45
4.1	Amount of datapoints on the different frequency bands throughout all the tests from both test devices	51
4.2	Frequency bands used by the MG21 during test 3 on 13.03 in each driving direction.	52
4.3	Highest frequency band used by MG21 in each section of the test area for both driving directions	53
4.4	Frequency bands used by the smartphone during test 3 on 13.03 in each driving direction.	54
4.5	Highest frequency band used by the smartphone in each section of the test area for both driving directions	55
4.6	Comparison between frequency bands used by the MG21 during round 1 and 3 from test 3 on 17.03 while driving north. NOTE: Kibana is set to display the minimum value if there are overlapping points on the map.	56
4.7	Comparison between frequency bands used by MG21 during round 1 and 3 from test 3 on 17.03 while driving south. NOTE: Kibana is set to display the minimum value if there are overlapping points on the map.	57
4.8	RSRP and RSRQ values from both the smartphone (red) and the MG21 (green) taken during test 1 13.03. The flat line in both figures between 09:15 and 09:30 is because of missing data points from a small break in the test.	60
4.9	Comparison between the best and worst RSRP values in each geographical position gathered from the MG21 throughout all tests driving south	62
4.10	The average RSRQ of the MG21 and the smartphone during the tests between 01.03 and 17.03	65
4.11	The average RSRP of the MG21 and the smartphone during the tests between 01.03 and 17.03.	67
4.12	The average RSRQ of the MG21 and the smartphone during the test conducted on 26.04	71
4.13	The MG21's RSRP before and after the switch to a capacity/higher frequency band (driving south)	73
4.14	eNodeB's that was used by the smartphone during all the tests. Skumsjoevegen (A), Lauvkampen (B), Vardalsaasen (C), Helset (D), Skumsjøen (E). Blue marking from A to B is the Test area. Taken from Google Maps	75
4.15	Possible handover occurring at the same time between the MG21 and the smartphone	76

4.16	The MG21's upload speed (Mbps) measured during the test on 01.03	77
4.17	Average upload speed for the MG21 for each frequency band	78
4.18	The MG21's upload speed, RSRP and frequency band for the first two tests on 17.03.	79
4.19	Example showing the comparison between the 800 MHz frequency band with good RSRP and the 1800 MHz frequency band with poor RSRP in the MG21's results, demonstrating better UL performance in the former.	79
4.20	The MG21's download speed from test 1 on 26.04	80
4.21	The MG21's average download speed per frequency band	81
4.22	The download speed, RSRP, and frequency band of the MG21 for the tests performed on 26.04.	82

Tables

3.1	Laptops used in the project	21
3.2	Signal bar threshold for RSRP and RSRQ, specific for Cisco Meraki gateways [68].	25
4.1	Tables containing general statistics	49
4.2	Comparison of RSRQ between the 800 MHz and 1800 MHz frequency bands for the test done between 01.03 and 17.03. The data is from both the smartphone and the MG21	64
4.3	Comparison of RSRP between the 800 MHz and 1800 MHz frequency bands from the tests on 01.03 to 17.03. The data is from both the smartphone and the MG21 (* -116 dBm was the lowest RSRP, but that captured during a possible handover. second highest was -106 dBm and there were several data showing this value) . . .	66
4.4	Comparison of RSRP between the 800 MHz and 1800 MHz frequency bands between the test done 26.04. Test data is from both the smartphone and the MG21	69
4.5	Comparison of RSRQ between the 800 MHz and 1800 MHz frequency bands between the test done 26.04. Test data is from both the smartphone and the MG21	69
5.1	Signal bar threshold for RSRP and RSRQ, specific for Cisco Meraki gateways [68]. (Reused Table 3.2)	86
5.2	Testing superparameter Proposition A with data from all tests, using the variables $X = Y = Z = 1$	89
5.3	Testing superparameter Proposition A with data from all test, using the variables $X = Y = 3$ and $Z = 1$	90
5.4	Testing superparameter Proposition A with data from all tests, using the variables $X = 7, Y = 2$ and $Z = 1$	90
5.5	Example of a superparameter test with data from all tests, using the first variant, with $RSRP_{Min} = -101, RSRQ_{Min} = -12, RSRQ_{Multiplier} = 3, Base = 90$	93
5.6	Example of a superparameter test with data from all tests, using the second variant, with $RSRP_{Min} = -101, RSRQ_{Min} = -12, RSRQ_{Multiplier} = 3, Speed_{Min} = 5, Speed_{Max} = 15, Base = 90$	95

- 5.7 Example of a superparameter test with data from all tests, using the third variant, with $RSRP_{Min} = -101$, $negRSRP_{Multiplier} = 10$, $RSRQ_{Min} = -12$, $RSRQ_{Multiplier} = 4$, $negRSRQ_{Multiplier} = 20$, $Speed_{Min} = 5$, $Speed_{Max} = 15$, $negSpeed_{Multiplier} = 15$, Base = 1000 96

Code Listings

C.1	Python script for combining logfiles	133
C.2	Python script used during testing	140
C.3	Python script used to create indexes and upload data to Elasticsearch	143
D.1	Docker compose file for demo containers	157
D.2	Python script for setting up the demo	159

Acronyms

- 3G** The third generation of cellular network technology. 5
- 3GPP** The 3rd Generation Partnership Project. 9, 50
- 4G** The fourth generation of cellular network technology. iii, v, 1–3, 5, 6, 8–11, 19–21, 23, 25–28, 31, 33, 35, 37, 40, 41, 43, 45–47, 58, 61, 68, 74, 85, 87, 88
- 5G** The fifth generation of cellular network technology. 3, 5, 8, 9, 11
- API** Application Programming Interface. 21, 22, 32, 34–38, 44, 58, 76, 82, 100–102
- CA** Carrier Aggregation. 10, 17, 23, 36, 47, 77, 80, 82, 86, 92, 100, 102
- CC** Component Carrier. 17
- CSV** Comma-separated values. 38, 45
- dB** Decibel. 12, 25, 49, 60, 62–64, 66, 69, 86, 88–91, 93, 95, 96
- dB_i** Decibel isotropic. 15, 24
- dB_m** Desibel-milliwatts. xvii, 11, 13, 25, 49, 60–62, 66, 69, 73, 74, 86, 88–91, 93, 95, 96
- DL** DownLink. 9–11, 17, 23, 63, 76, 80, 82
- eNodeB** Evolved Node B. xiii, xiv, 6, 11, 12, 14, 17, 23, 25–29, 35, 36, 39–41, 43, 47, 52, 58, 68, 74–77, 80, 82, 100
- FDD** Frequency-division duplexing. 10, 11
- GHz** Gigahertz. 6–8, 78
- GPS** Global Positioning System. 20, 22, 31–33, 36
- GSM** Global System for Mobile. 11

- GUI** Graphical user interface. 34
- HTTPS** Hypertext Transfer Protocol Secure. 22
- IoT** Internet of Things. 1, 2, 16, 20
- ITU** International Telecommunications Union. 5
- JSON** JavaScript Object Notation. 22
- KHz** Kilohertz. 6, 7
- LTE** Long-Term Evolution. 5, 6, 8–12, 14, 17, 20, 21, 23, 25, 27, 28, 88
- Mbps** Megabits per second. xv, xxv, 4, 5, 21, 49, 66, 76, 77, 79, 80, 82, 86, 89, 90, 93, 95, 96
- MG21** Cisco Meraki Cellular Gateway 21. xiii–xv, xvii, 20, 21, 23–25, 31, 34–43, 45–47, 49–60, 62–74, 76–83, 86, 89–93, 95, 97, 99–102
- MHz** Megahertz. xiii, xv, xvii, 3, 7, 9–11, 17, 23, 24, 27–30, 39–43, 45, 46, 50, 52–58, 63, 64, 66–70, 72–75, 77–79, 81, 82, 88–90
- ms** millisecond. 13, 14
- Nkom** The Norwegian Communications Authority. 9
- NR** New Radio. 8, 9, 11
- NTNU** Norwegian University of Science and Technolog. vii, 2, 3
- PIFA** Planar Inverted-F Antennas. 24
- QoS** Quality of Service. 12, 14, 41
- RSRP** Reference Signal Received Power. xiii–xv, xvii, xxvi, 11–13, 20, 21, 23, 25, 26, 29, 30, 38, 41–43, 46, 47, 49, 53, 58–63, 66–69, 72–74, 78, 79, 81–83, 85–96, 100–102
- RSRQ** Reference Signal Received Quality. xiv, xvii, 11, 12, 20, 21, 23, 25, 41–43, 46, 47, 49, 53, 58–66, 68–72, 74, 82, 83, 85–96, 100–102
- TDD** Time-division duplexing. 10
- TEMS** TEst Mobile System. xiii, 25–30, 39, 40, 50, 53–55, 74, 75, 82, 102
- TTT** Time-to-Trigger. xiii, 13

UE User Equipment. 6, 7, 9–14, 16, 17, 22, 25, 27–29, 36, 39, 41–43, 46, 51, 58, 61, 68, 72, 74

UHF Ultra High Frequency. 7

UL UpLink. xv, 9–11, 17, 63, 76, 79, 80, 82

Glossary

Antenna A specialized device that converts electric current into radio waves and vice versa. xiii, xxv, 14–16, 23, 24, 40, 55, 58, 59

Antenna radiation How the antenna radiates electromagnetic energy into space and receives electromagnetic energy waves. 40, 42, 50, 55, 82

Bandwidth The maximum amount of data that can be transmitted over a network connection per time unit, calculated in Megabits per second (Mbps). 4, 7–11, 17, 23, 32, 40–42, 45, 46, 76, 85–87, 91–95, 101, 102

Capacity band A frequency band that gives high network capacity, but at the cost of low coverage. 8–11, 28, 42, 50, 51, 53–55, 68, 78, 81–83

Cellular gateway A device connects it self and other devices to a cellular network. 17, 72

Coverage band A frequency band that provide high coverage, but at the cost of low network capacity. 7, 8, 10, 29, 42, 50, 64, 68, 78, 81, 83

Directional antenna A type of antenna that concentrate the radiation energy in specific directions. 16, 17

Dmnidirectional antenna A type of antenna that radiates and receives radio waved equally in all directions in the horizontal plane. 16, 24

Frequency band A range of frequencies that is allocated for Telecommunication. xiii–xv, xvii, xxv, 3, 7–11, 17, 20, 21, 23, 25–29, 34–43, 45–47, 50–58, 63, 64, 66–70, 72–75, 77–83, 85, 87, 88, 92, 101, 102

Gain The radiation power in a particular direction compared to an isotropic antenna.. xiii, 15, 16, 24, 55

Handover Process of when a mobile device switches between base stations. xiii, xiv, xvii, 12–14, 28, 29, 40, 41, 45, 58, 66, 74, 76, 77, 80, 82

Hysteresis margin A value that defines how much the RSRP of a neighbouring cell must be over the serving cell before a handover is initiated. xiii, 12, 13

Interference Disruption or disturbance that affect the quality of a radio signal, caused by unwanted signals and/or physical obstacles. 7, 9, 10, 12, 13, 28, 36, 42, 59, 61, 63, 72, 86, 100–102

Network congestion Reduced network performance caused by an excess of data traffic. 7, 37, 58

Signal bar Strength of the received signal a devices gets from a network. xvii, 25, 61, 74, 86, 89–91, 93–97, 101

Chapter 1

Introduction

The fourth generation of cellular network technology (4G) has been used for cellular connections for more than a decade and is supported by most cellphones produced after it was introduced. With the growth of Internet of Things (IoT) appliances, the use of 4G for devices other than cellphones has increased. The increasing demand for 4G connectivity and the requirement of continuous connection for some devices have led to an increased need to map out the coverage, even outside of cities and residential areas. One such use case is automated construction vehicles that depend on 4G connection to operate. If they lose the connection at any point, they will typically halt and enter safe mode. In quarries, the connection can change depending on the depth of the construction over time, leading to areas where the connection is completely lost. Therefore, it becomes important to have updated information about the 4G coverage in all areas of the quarries.

1.1 Background

This bachelor project is based on Telenor's task of supplying 4G cellular connection to quarries. The quarries use 4G to operate their automated tip-trucks, and they require a constant connection, otherwise the trucks will enter safe mode and stop. 4G is used to make the operation safer and more flexible by enabling the trucks to share their positions. One particular quarry has dug so deep that it is now outside the cellular coverage area of the nearest base station. This happened slowly over time, but there is no warning that goes off before the connection is completely lost. One of the objectives (Section 1.2.1) of this project is to create a superparameter that can be used to trigger warnings at certain thresholds, so that countermeasures can be implemented before the 4G connection is completely lost.

1.2 Assignment

The assignment (Appendix A) given by Telenor is to create a method for testing the 4G network coverage. By using this method on a given section of road, data can be collected to create a superparameter that IoT devices, like the aforementioned trucks, can monitor to determine if they are in danger of losing their 4G connection.

1.2.1 Objectives

The objectives for this project are as follows:

- **Create a test setup:** Develop a test setup for testing the 4G connection using the provided equipment.
- **Display test data in a heatmap:** Present the test results neatly on a map.
- **Create a superparameter:** Develop a superparameter that can indicate "downtime" for a given use-case and provide warnings before the 4G connection is lost.
- **Create a playbook:** Create a playbook containing a short presentation, a detailed approach to deliver a demonstration and a video that can be used as a backup for the demonstration.

1.3 Target Audience

This project and thesis are intended for the telecommunication industry, specifically Telenor. In addition, it is intended to be useful for anyone who is using devices that require a stable 4G connection to operate, like the aforementioned vehicles in the quarries. The thesis assumes that the reader has some experience or knowledge within informatics to fully comprehend and make use of it.

1.4 Our starting point

Our team comprises four bachelor students, all pursuing the same field of study: Bachelor's in Digital Infrastructure and Cyber Security at Norwegian University of Science and Technology (NTNU). Within the team, there is a collective background in various network courses as part of our bachelor's degree. Additionally, experience in programming languages such as Python and C has been instrumental in developing tools for testing. Moreover, hands-on experience with Docker has been utilized to set up the system for analyzing and visualizing the test data.

1.5 Roles outside the team

Outside of the team itself there have been two significant contributors:

- **Telenor:** Telenor is our client who has given us our assignment. Specifically our contacts with Telenor go through Henning Elvestad, Bjørn Isene and Andreas Rømo.
- **Supervisor:** The team has been assigned a supervisor by NTNU to oversee the project and provide guidance, ensuring the delivery of something that NTNU considers a good thesis. The supervisor assigned to our team is Ernst Gunnar Gran.

1.6 Scope

To be able to do a good job on the assignment, it is important to establish a clear scope to not stray too far from the objectives.

All of the testing has been done within the section of road designated by Telenor. The testing has focused solely on 4G because 4G is used in the use-case mentioned earlier in Section 1.1. Additionally the hardware provided by Telenor does not support The fifth generation of cellular network technology (5G). 4G is also still very relevant and widely used today.

The tests has focused mainly on the 800 Megahertz (MHz) and 1800 MHz frequency bands (which will be explained in Section 2.2). This is to focus on a single low-band spectrum band and a single mid-band spectrum band for simplicity (explained in Section 2.1.6). 5G is not a part of this project and will only be briefly mentioned here and there.

The tests have been conducted within a section of a road chosen by Telenor. The particular test area was chosen due to previous signal readings showing inconsistency in 4G connectivity in that area, which provided more data on various signal data. The tests has been done by driving through the area in a car with the equipment placed inside the car.

1.7 Thesis Structure

This thesis is inspired by the IMRaD structure, which stands for Introduction, Methods, Results and Discussion [1]. In addition there will be a theory, superparameter and conclusion chapter.

- **Chapter 1 - Introduction** will introduce the thesis, with the main background to the thesis, assignment given from Telenor and the thesis scope.
- **Chapter 2 - Theory** will give the reader an explanation of concepts that is needed to understand the results and the discussion.
- **Chapter 3 - Methods** will describe how the different tests were executed, as well as a description on the software and hardware used in the testing. A description of the test area and previous signal measurement readings from the test area. Finally, a hypothesis was formulated.
- **Chapter 4 - Results and Discussion** showcases the results gathered from each test scenario and discuss them.
- **Chapter 5 - Superparameter** will discuss what the superparameter will look like and some concrete proposals.
- **Chapter 6 - Conclusion** of the thesis based on the discussion, learning achieved from doing the project and further work that can be done.

1.8 Notes

Before the rest of the thesis begins, there is some clarifications:

In this thesis, *bandwidth* refers to the maximum amount of data that can be transmitted over a network connection per time unit, calculated in Megabits per second (Mbps) [2].

All the figures and tables used in this thesis are created by the team members, except the figures and tables where it is specified in the caption who the copyright owner is.

Chapter 2

Theory

The theoretical part will cover some basic theory around frequencies, signal, 4G Long-Term Evolution (LTE), measurement of 4G quality, radio propagation and other relevant theory related to the project, aimed to give the reader a better understanding of the main concepts discussed later in the report.

2.1 4G

4G is probably the most important concept to understand in this project, as it is the focus of the entire project: to measure 4G coverage in a specified area and create a superparameter using data from the tests. 4G is short for "4th Generation" and as the name implies it is the 4th generation of mobile data technology [3]. 4G is considered the successor of The third generation of cellular network technology (3G). 4G is designed to accommodate a much larger volume of devices and more data-heavy activities like streaming. 4G is a type of cellular network that are often used in mobile devices [4]. The successor of 4G is 5G - "5th Generation". While 5G is newer and in many cases faster than 4G, expanding 5G infrastructure is still an ongoing process. Therefore the 5G coverage is limited compared to 4G. More on cells and cellular networks in the next few paragraphs.

2.1.1 4G Long Term Evolution

When 4G was introduced as a concept by International Telecommunications Union (ITU) in 2008 [4], they set a minimum requirement for 4G download speed at 100 Mbps, a speed that was unheard of at the time. Consequently, many telecommunications companies could not reach the requirement to label what they provided as 4G, despite using 4G technology. Therefore they called it 4G LTE, which basically means that they use 4G technology, but are not at the minimum requirement yet, but working toward achieving it [3].

2.1.2 Base stations in 4G LTE

Like all wireless networks, 4G LTE networks work by using radio waves to transmit and receive data. The radio waves are transmitted from and received by base stations that connect devices to the rest of the mobile network. These base stations are what makes such wide coverage of wireless networks feasible [5].

The type of base station used in mobile and cellular networks utilizing LTE technology is called Evolved Node B (eNodeB) [6]. A base station provides cellular coverage to one or more geographical areas known as cells. These cells often overlap, and the same User Equipment (UE) can be in several cells at the same time. UE refers to devices that are used in LTE and 4G wireless communication by an end user to access the 4G network. The cell to which a UE is connected is known as the serving cell, and the other cells in which the same UE is present, but not currently connected to, are known as neighbouring cells [7]. An eNodeB can be seen in Figure 2.1.



Figure 2.1: Example on what an eNodeB looks like

2.1.3 Radio Waves

Radio waves are a type of electromagnetic radiation with long wavelengths and low frequencies. They also have a low amount of energy and are therefore a safe form of electromagnetic radiation for humans. Radio waves can have frequencies ranging from 3 Kilohertz (KHz) up to 300 Gigahertz (GHz) [8]. Radio waves can pass through many different materials, such as wood, dirt and stone, but not all ma-

materials, for example, they cannot penetrate iron. The reason they can easily pass through so many materials is due to their long wavelengths [9]. The lower the frequency, the longer the wavelength [10]. Furthermore, low frequencies correspond to lower potential bandwidth, and vice versa for high frequencies [11].

2.1.4 Interference

Interference occurs when a signal is disrupted by something. When dealing with radio waves going through the air, there are many potential sources of interference. However, for this project, the only truly relevant interference comes from physical obstructions. Some obstructions, like metal, reflects radio waves, while others, such as glass and water, allow the radio waves to pass through while also reflecting some of them. These reflections still cause some interference [12].

2.1.5 Network congestion

Network congestion occurs when an excess of data traffic leads to reduced network performance for the end user. This happens because there is a limit on the bandwidth for each segment of the path that data must travel to reach its destination [13]. Thus, the more users there are on a network, the more likely they are to experience reduced performance due to network congestion.

2.1.6 Frequency Bands

Although the interval of radio waves is large (3-300.000.000 KHz), the interval that has been traditionally used for mobile communication is smaller, the so called Ultra High Frequency (UHF) band that ranges from 300 MHz to 3 GHz [14, p.7]. The reason for this is that the higher the frequency, the higher the bandwidth, but the coverage area gets smaller with higher frequencies. The UHF band has a good balance between bandwidth and range. The lower parts of the UHF band is suitable for transferring data over long distances and through obstacles like hills, woods and buildings (Section 2.1.3), while the upper parts of the UHF band are well-suited for providing services that require faster data transfers. Generally speaking, frequency bands with a frequency of less than 1 GHz are considered good for coverage, while frequency bands with a frequency over 1 GHz are considered good for capacity.

Frequency bands are divided into different categories based on the frequency the frequency bands provide. These categories are [15]:

- **Low-band spectrum band:** These frequency bands operate on frequencies lower than 1 GHz on the radio spectrum chart, and are often referred to as "coverage bands" due to their expansive coverage area. The carrier signals can easily penetrate through walls and windows, but offer low bandwidth capacity. A majority of UE utilize the low band, with rarely any connection

issues to the mobile network. During high-demand periods, 4G LTE capacity may be limited within these frequencies, potentially resulting in user slowdown during high traffic hours. coverage bands are used in both 4G LTE and 5G New Radio (NR) communications.

- **Mid-band Spectrum band:** These frequency bands, particularly in the 1-2.6 GHz range, offer higher frequencies for 4G LTE communication. Often referred to as "capacity bands", these provide greater capacity and a good coverage range. They offer higher speed than low-bands and better coverage than high-bands [16]. Although mid-bands can penetrate walls and buildings, their penetrative capabilities are not as strong as those of the low band. These frequency bands offer high capacity and low latency. Capacity bands are also used in 4G LTE and 5G NR communications.
- **High-band spectrum band:** These frequency bands have very high frequencies in the 6 GHz and above range. They provide very high capacity and very low latency, but their coverage is significantly limited. These frequency bands are exclusively used in 5G NR communications.

Figure 2.2 illustrates that achieving higher coverage comes at the expense of a lower bandwidth in the lower frequency bands, while higher frequency bands offer users a better bandwidth and user experience, albeit at the cost of reduced coverage.

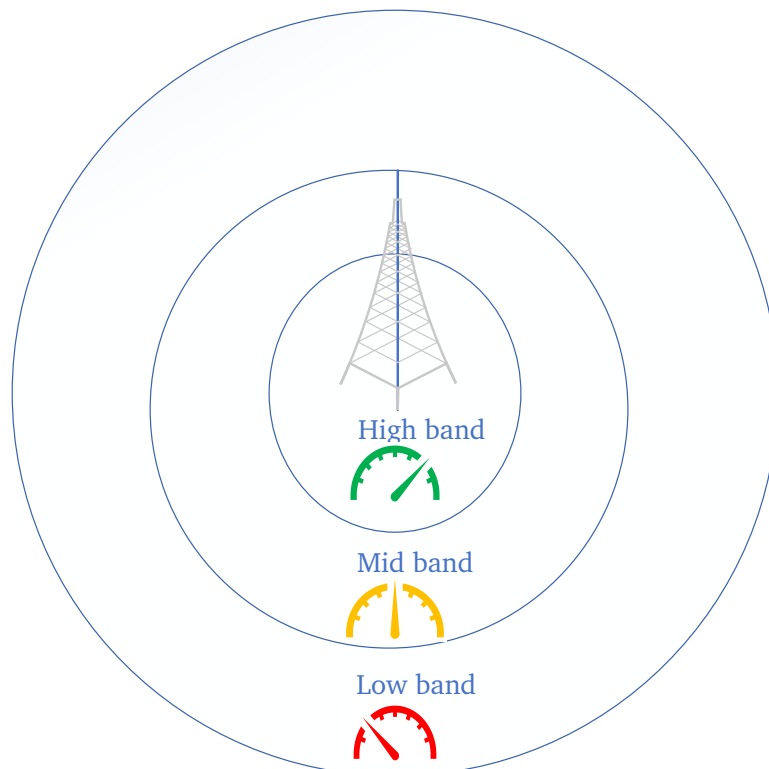


Figure 2.2: Illustration of the relation between high, mid, and low bands

To deliver effective capacity and coverage for 4G LTE mobile communication nationwide, a combination of coverage and capacity bands is necessary. There might be areas where multiple frequency bands have overlap cells. These frequency resources can be leveraged to achieve good bandwidth [14, p.16-17].

2.2 3GPP frequency bands

The 3rd Generation Partnership Project (3GPP) is an umbrella term for organizations that establish standards regarding the design of frequency bands and other telecommunication standards [17]. These standards includes which frequencies within a frequency band that are used for UpLink (UL) and DownLink (DL), which UE utilizes for data transmission and reception, respectively. They also determine the supported channels bandwidth within a frequency band. In addition, 3GPP provides other important technical information beneficial to telecommunication operators when building and expanding their networks to meet all user needs [14, p.53].

The 3GPP frequency bands utilized within the Norwegian public telecommunication network include 450 MHz, 700 MHz, 800 MHz, 900 MHz, 1800 MHz, 2100 MHz, 2600 MHz and 3600 MHz. There are over 1000 frequencies allocated for telecommunication technologies such as LTE (4G) and NR (5G). With the ongoing development of 5G NR, an even greater number of frequencies will become available for public telecommunication [14, p.12-13].

The Norwegian Communications Authority (Nkom) is responsible for administering the public regulations for telecommunication, frequency management and spectrum allocation for Norwegian operators [18]. Frequency resources are allocated through a frequency auction. Telenor, for instance, has been allocated frequencies within all available frequency bands in the Norwegian public telecommunication network used for 4G LTE, except the 450 MHz frequency band [14, p.9].

To thoroughly understand these frequency bands, it is necessary to first to gain an understanding of duplexing, followed by a detailed explanation of the frequency bands used in this thesis.

2.2.1 Duplexing

When a user needs to transmit some form of voice or data, there are different methods available for how this communication can be carried out. Implementing such a method is necessary to establish communication between the transmitter and receiver, ensuring smooth transmission with minimal interference [19, p.461-462].

There are three primary communication modes: simplex, half-duplex, and full-duplex [19, p.461]. Most of the frequency bands in Norwegian telecommunication use full-duplexing. Full-duplexing allows both parties involved in the communication to transmit and receive information simultaneously, enabling real-time two-way communication, such as a phone call [19, p.462].

Two duplexing schemes are predominantly used in wireless communication: Frequency-division duplexing (FDD) and Time-division duplexing (TDD) [19, p.462]. However, in the frequency bands found in the test area for this thesis, only FDD duplexed frequency bands will be encountered.

In FDD, there are two separate frequency bands: one for UL and another for DL. The user is allocated one frequency for UL, allowing them to transmit data or voice to a base station, and another frequency for DL, enabling them to receive data or voice from the base station. This facilitates simultaneous two-way communication [19, p.462-464].

To separate the UL band and DL band, a Guard band must be implemented. A Guard band is a narrow frequency range that separates the UL and DL bands. Its purpose is to ensure simultaneous communication by preventing interference between the two carrier frequencies [20].

2.2.2 800 MHz frequency band (B20)

The B20 frequency band, also referred to as the 800 MHz frequency band, is used in Norwegian telecommunication to access the 4G LTE network. 800 MHz is a coverage band that uses FDD as its multiplexing method. As mentioned earlier, a coverage band has good penetration through buildings/objects and covers large areas, but the bandwidth of a coverage band is not as good as that of a capacity band. Here are the frequencies that 800 MHz can operate on [14, p.28-29]:

- 791-821 MHz on the DL
- 821-832 MHz Guard band
- 832-862 MHz on the UL

In Norway, Telenor have been allocated these frequencies within the 800 MHz frequency band, which Telenor can use for the public telecommunication [14, p.28-29]:

- 801-811 MHz on the DL
- 842-852 MHz on the UL

UEs that supports LTE-Advanced and LTE-Advanced pro can deploy carrier aggregation with 800 MHz, combining it with carriers in other frequency bands or within the same frequency band. Carrier Aggregation (CA) is possible between 800 MHz and 1800 MHz frequency bands [21].

2.2.3 1800 MHz frequency band (B3)

In Norwegian telecommunication, the B3 frequency band, also referred to as the 1800 MHz frequency band, was previously used to provide users access to the Global System for Mobile (GSM) network before the introduction of LTE. Today, the 1800 MHz frequency band provides LTE connectivity to users. It also provide users access to the 5G NR network for devices that support 5G [14, p.34-35].

1800 MHz is a capacity band, providing acceptable coverage with its ability to penetrate through walls and offering good coverage in urban areas and cities, as well as providing great bandwidth capacity [22].

The frequencies allocated for the 1800 MHz, which uses FDD as its duplexing method, are as follows:

- 1710-1785 MHz on the UL
- 1785-1805 MHz Guard band
- 1805-1880 MHz on the DL

In Norway, Telenor have been allocated these frequencies within the B3 frequency band, which Telenor can use for public telecommunication [14, p.34-35]:

- 1715-1745 MHz on the UL
- 1810-1830 MHz on the DL

2.3 4G signal level measurement

When your mobile phone or any other UE needs to connect to a 4G LTE network, the device needs to establish a connection to the cell that provides the strongest and highest quality signals which the UE can receive. In an LTE network, the UE primarily uses two parameters to measure the signal it receives from a cell: Reference Signal Received Power (RSRP) and Reference Signal Received Quality (RSRQ) [23].

2.3.1 RSRP - Reference Signal Received Power

RSRP is a parameter that measures the signal strength a UE receives from eNodeBs in a 4G LTE network. An eNodeB will frequently send a cell-specific reference signal to the UE, and the UE will measure the signal strength based on that reference signal. This reference signal, sent frequently by the eNodeB, is dedicated specifically to signal measurement. A UE can measure the RSRP not only from its serving cell but also from several neighbouring cells. RSRP is measured in negative dBm, with a range of -44 dBm to -140 dBm. Thus, the higher the RSRP value, the stronger the signal from the cell. A higher RSRP ensures a more reliable and faster connection to an eNodeB. RSRP is a decisive factor to consider when a UE

is performing a cell selection to choose the cell with the best signal. The UEs will measure the RSRP of the neighbouring cells, and if the RSRP is better than the current serving cell, a cell selection will be performed [23].

2.3.2 RSRQ - Reference Signal Received Quality

RSRQ is a cell-specific parameter that is used to measure the quality of the signal received from a specific cell. It is similar to RSRP in that both are used to rank the quality of signals received from eNodeB. RSRQ considers RSRP and all interference affecting the signal. A high RSRQ value indicates less interference are affecting the signal, as well as a higher value of RSRP. RSRQ is measured in Decibel (dB). The closer the RSRQ is to 0 dB, the better the quality of the signal. RSRQ values typically range between -3 dB and -19.5 dB. RSRQ is also used when making cell selection for UEs, serves as an additional parameter when RSRP does not provide sufficient information to perform the cell selection [24].

2.4 Handover

Handover is the process in telecommunication where cellular devices switch from their serving base station to a neighboring base station that has a better signal for the cellular device. Cellular devices are often not stationary, making it necessary for them to move from one base station to another in order to maintain connectivity to the cellular network [25].

UEs connected to the LTE network frequently measure RSRP and other signal measurements parameters from their serving cell and all other neighboring cells. The UE then sends these measurements to its serving cell to verify that the value of the serving cell is stronger than that of other cells, ensuring that the UE maintains a strong enough signal for connectivity to the LTE network and receives the desired Quality of Service (QoS) [26, p.537-538].

If a UE is moving away from the serving cell and the RSRP value becomes lower than that of a neighboring cell, it does not necessarily mean that the UE will handover to the eNodeB with a better RSRP value. This is because there are other factors to consider before initiating the handover [26, p.537-538]. These factors will be introduced in the following subsections.

2.4.1 Hysteresis margin

The hysteresis margin is an important parameter in the handover process. It is a value that defines how much the RSRP of the neighbouring cell must be over the RSRP of the serving cell before initiating a handover. For example, if the RSRP

of the serving cell is -90 dBm and the hysteresis margin is 3 dBm, then the RSRP of the target cell should be above -87 dBm to initiate the handover process. Hysteresis margin is useful in the sense that the UE can avoid performing many unnecessary handovers just because the RSRP is going up and down slightly due to interference or moving closer to the edge of the cell. Hysteresis margin is there to ensure the UE gets a stable and high-quality connection before switching cells [26, p.542-543].

2.4.2 Time-to-trigger

To initiate the handover process a Time-to-Trigger (TTT) must be present. TTT means that the RSRP of the neighbouring cell needs to be stronger than the RSRP of the serving cell + hysteresis margin within a specific time interval before triggering the handover process for the UE. TTT contributes to reducing the number of handovers being performed, by ensuring that signal strength is stable enough in the neighbouring cell before the handover is initiated. With TTT the UE can avoid a ping-pong handover, which is handover back and forth between the same cells in short periods [26, p.543]. Figure 2.3 illustrates how the handover procedure is done. TTT as well as the hysteresis margin are set by the network operators, so the values can vary from each operators own network requirements. In the figure the TTT is set to 100 millisecond (ms), after 100 ms the handover is initiated.

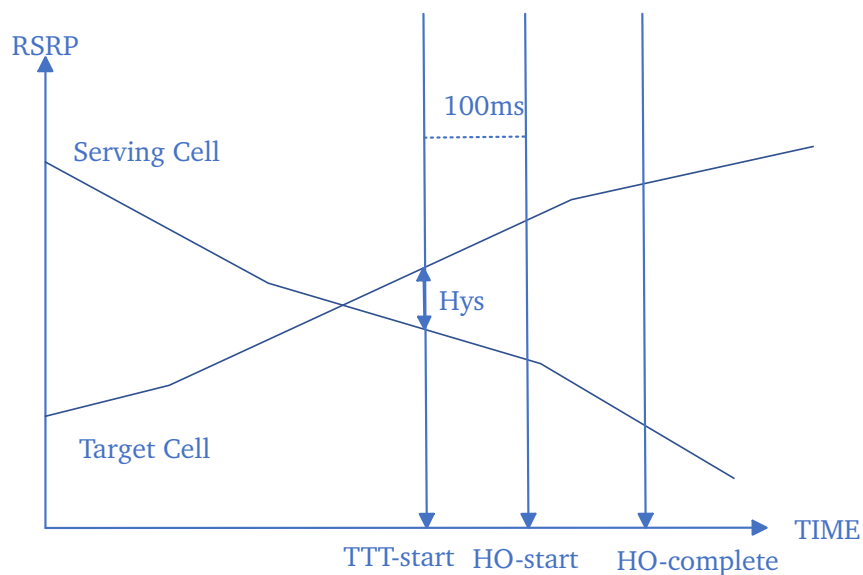


Figure 2.3: Illustration of the handover process, HO=Handover, Hys=Hysteresis margin TTT=Time-to-trigger. Inspired by Telecommunication Systems [26, p.542]

2.4.3 Handover interruption time

During the handover process, there may be some problems that can affect the user experience on the LTE network, such as latency, data not transmitting, or the inability to establish a connection to the target cell. One of the issues encountered during the handover process is the handover interruption time.

Handover interruption time refers to a short period during the handover process when neither the eNodeB nor the UE can transmit data to each other. In LTE, the handover interruption time typically ranges from 30 to 60 ms, but it can be longer depending on environmental conditions that affect radio signals [27]. This handover interruption can impact QoS and the user experience, especially when the network is performing unnecessary handovers. Therefore, it is crucial to perform handovers only when necessary and in the most effective manner to reduce latency and provide the best possible connection for the UE.

2.5 Antennas

UEs and base stations need to transmit and receive electromagnetic waves to access the mobile network. But how do UEs and base stations send these electromagnetic waves? Every UE contains a small electrical device called an antenna. The antenna converts electric power, representing data from the UE, into radio waves, which can be transmitted as signals to the base station. The antenna can also convert radio waves sent from base stations into data that the UE will process [28].

2.5.1 Antenna radiation pattern

Every antenna has its own radiation pattern. A radiation pattern refers to how the antenna radiates electromagnetic energy into space and receives electromagnetic energy waves [29, p.2-3].

Antennas radiate in all directions, so it is normal to showcase the radiation pattern in a 3D figure. It is also common to illustrate the radiation pattern in a principal plane pattern. The principal plane pattern shows how the antenna radiates in two perpendicular planes: a horizontal plane (xy in Figure 2.4) and a vertical plane (xz and/or yz in Figure 2.4), using two or three 2D figures. This provides a better understanding of how the radiation occurs in each direction [29, p.2-3]. The radiation pattern for an antenna on a base station showcases how the radiation is distributed in all directions and which direction provides the best coverage for the UE. The same can be said for the radiation pattern of a UE by showcasing the best direction in which the antenna radiates, allowing the UE to establish a communication link with a base station.

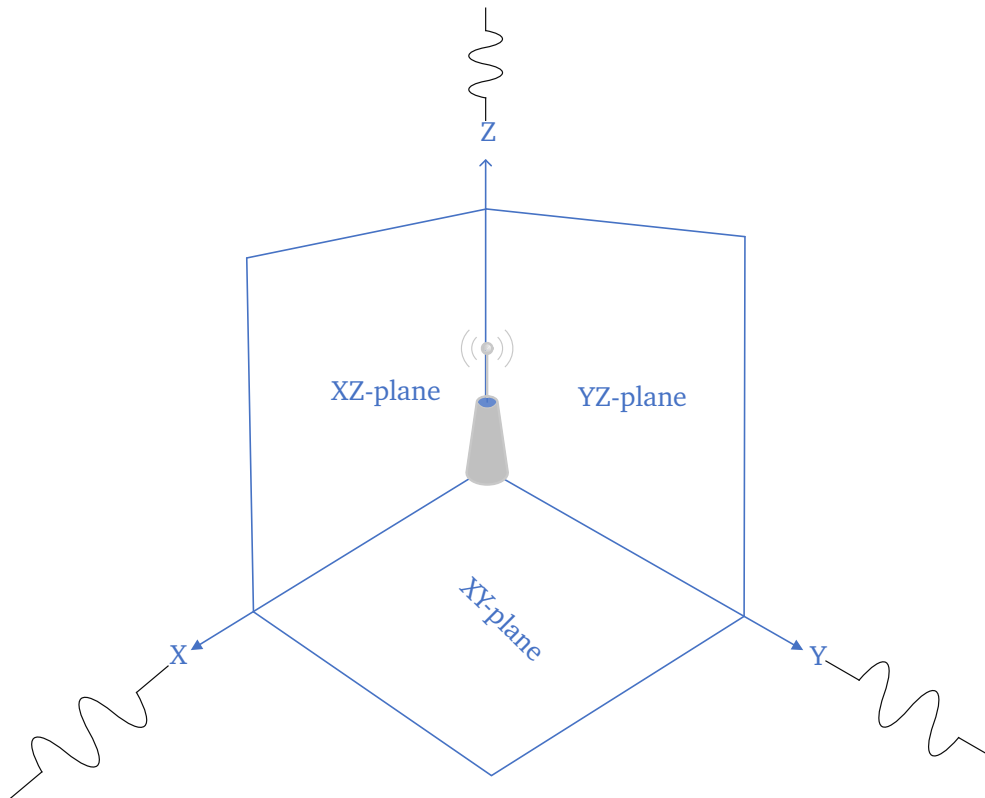


Figure 2.4: XYZ-plane radiation illustration. Inspired by Wikipedia [30].

2.5.2 Gain and types of antennas

Gain, in the context of an antenna, is the radiation power in a particular direction compared to an isotropic antenna in the same direction [29, p. 3-4]. An isotropic antenna is an ideal antenna that radiates in every direction with the same intensity [31]. An isotropic antenna, therefore, does not have any gain in any direction. Gain is measured in decibels relative to an isotropic radiation or Decibel isotropic (dBi). The higher the gain is in one direction, the more concentrated the radiation is in that particular direction. The signal will reach a longer distance, but the coverage area of the signal is smaller. Figure 2.5 showcases how low gain antennas covers a larger area, while high gain antennas will reach longer distance with less area covered [32].

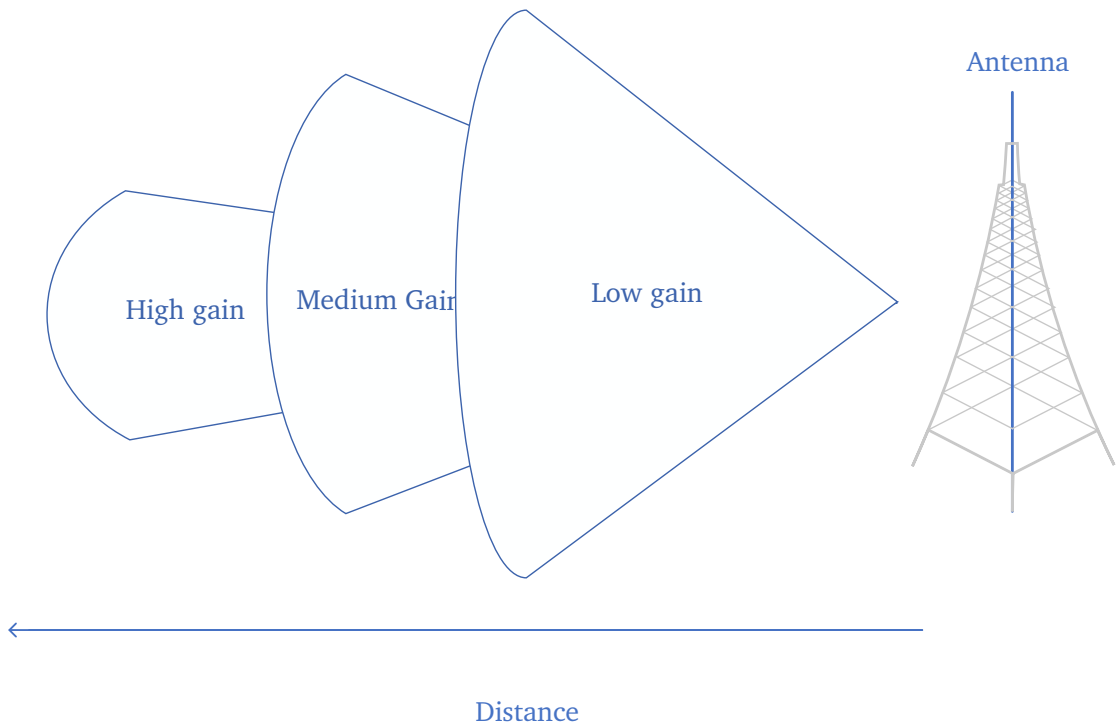


Figure 2.5: Antenna gain illustration. Inspired by Inhandnetworks [32]

High gain is preferable when the direction of the receiving and transmitting antennas is known to each other at any given time. However, since the UEs aren't always stationary and their orientation and position can vary significantly, it is better for mobile antennas to have low gain [33].

Omnidirectional antennas are a type of antenna that radiates and receives radio waves equally in all directions in the horizontal plane. This is due to the low gain found in these types of antennas. omnidirectional antennas are mostly found in devices that operate on radio frequencies, such as mobile phones and IoT devices [34].

The best use-case for omnidirectional antennas is when good coverage in all directions is required, such as in suburban and urban areas where there are many users who need cellular coverage [35].

Directional antennas, in contrast to omnidirectional antennas, are antennas that concentrate the radiation energy in specific directions rather than equally in all directions. In simple terms, they exhibit gain in particular directions [29, p.3-4]. Directional antennas are employed in areas where the intention is to focus the radiation within a specific region [36].

Since the radiation is concentrated in a more specific area, they are able to reach longer distances and provide a stronger signal in this area. For base stations, directional antennas are more suited to rural areas such as the countryside where the distance between the UE and the base station can be long. The directional antenna can be pointed to these rural areas to provide them with coverage [35].

2.6 Carrier aggregation

Carrier Aggregation (CA) is a technique that is used in LTE-advanced networks to increase the bandwidth and improve the network performance in the UL, DL or both. The concept of CA is to combine multiple LTE carrier signals into an aggregated signal so that the receiving UE gets increased bandwidth [37, p.4-6]. This requires the UE to be within an area that has overlapping cells to make the aggregation of each carrier possible [38, p.842-843].

Each of these LTE carriers that is participating in the CA is called a Component Carrier (CC). Each of these CCs can have different bandwidth from each other and each CC can have a bandwidth of 1.4, 3, 5, 10, 15 or 20 MHz. CA in LTE-advanced can combine up to five CCs for UL and DL to be aggregated together which allowed for a maximum bandwidth of 100 MHz to be transmitted to a user [39]. Component Carrier (CC) can come from one or several frequency bands, and UE's also can get Component Carrier (CC) from one or several eNodeB's. Again this requires cells from both eNodeB's to have cells that are overlapping [40].

2.7 Cellular gateway

A cellular gateway is a device that serves the purpose of gathering data from devices connected to it via various protocols, converting the data from these devices into an IP-protocol. This allows the cellular gateway to send and receive data from the cellular network. Devices that don't support connectivity to the internet via a cellular network, will take advantage of a cellular gateway to gain access to the internet [41]. For example, a laptop can be connected to a cellular gateway via Ethernet or any protocol supported by the particular cellular gateway. The cellular gateway will then provide internet connectivity to the laptop from the cellular network.

Chapter 3

Methods

This chapter will go in-depth on the test setup used during the testing phase of this project, describing all the hardware and software employed during the tests. Furthermore, it will provide a thorough description of the test area used and analyze the previous signal measurement readings conducted by Telenor. Finally, hypotheses will be formulated to predict the test results.

Figure 3.1 shows a simplified version of how the tests will be performed. A car will be driven from point A to B in a designated test area. Inside the car, various equipment will be placed to perform network measurements of Telenor's 4G network.



Figure 3.1: Simplified figure of how the tests were performed.

3.1 Hardware

The following sections will describe the hardware used in the test setup.

3.1.1 Cisco Meraki cellular gateway (MG21)

The Cisco Meraki Cellular Gateway 21 (MG21) [42] will be used in this project, and this device is provided by Telenor. Cisco [43] is a company best known as one of the biggest companies in the world that develops computer networking systems and networking equipment. The MG21 is a Cisco Meraki product, which differs slightly from a regular Cisco device. A Cisco Meraki product is managed and configured remotely via Meraki cloud (more on Meraki cloud in section 3.2.1) [44].

MG21 is also capable of taking signal measurement from the network it is connected to and measure the quality of the connection to the 4G LTE network. The MG21 takes measurements such as RSRP, RSRQ, and frequency band information [45]. This is a vital feature of the MG21 used to measure the quality of Telenor's 4G LTE networks. The MG21 is a device that is designed to be stationary, either in an outdoor or indoor environment [46].

3.1.2 Nordic Thingy 91

The Nordic Thingy 91 is a Multi-Sensor cellular IoT prototyping platform manufactured by Nordic Semiconductor. IoT is a concept based on connecting any type of electronic device to the internet, enabling them to communicate with other connected devices [47]. The Thingy 91 contains a multitude of different sensors, including temperature, air pressure, and motion. By using the included SIM card, it can connect to cellular networks. The primary intended function for this device was Global Positioning System (GPS) positioning. Another feature was the ability to read signal values from the cellular connection and compare them with the values from the MG21. Through C programming, it is possible to select the desired features from the Thingy 91, customize their functionality, and determine the desired output [48].

3.1.3 Smartphones

A smartphone have been used throughout the tests to run various applications. These applications have been used to take signal measurement readings from Telenor's 4G LTE network and capture Global Positioning System (GPS) readings. The smartphone used in the tests has been a Google Pixel 6 Pro and its mobile operator is Telenor.

3.1.4 Laptops

Laptops were also used throughout the test scenarios to transmit and receive data through Telenor's 4G LTE network, where the laptops were connected to MG21. Table 3.1 contains the laptops that were available with this project:

	Model name	Model Nr
1	Asus ROG Strix G15	G513RW-HQ173W
2	MSI GF63 Thin 15,6	11SC-687NEU-BB51140H8GXXDX11M
3	Microsoft Surface Laptop	D9P-00001
4	Lenovo Yoga C940	81Q90008MX

Table 3.1: Laptops used in the project

3.1.5 Remote Server

A remote server running Unraid [49] was used during the project. This server was mostly used to test the download and upload speeds of the MG21 during testing. The server was on a stable 600/600 Mbps internet connection.

3.2 Software and tools

Various software and tools have been used throughout the project to gather data, configure hardware, process, and visualize the data gathered from the tests. The following sections will explain the software and tools that have been used in this project.

3.2.1 Cisco Meraki cloud

MG21 is managed and monitored in Cisco Meraki cloud [50]. The Meraki cloud has a web-based dashboard that enables administrators to remotely configure the MG21 device, and view device data at any given time such as RSRP, RSRQ, which frequency band is used, and much more [50].

During testing, it was observed that the signal measurement data from the Meraki cloud updates almost in real-time, with a 1-2 second interval for data updates in the cloud.

One inconvenience related to the Meraki dashboard is that it does not provide the ability to display information about carrier aggregation occurrences [51, p.6].

3.2.2 Cisco Meraki RESTful API

The Cisco Meraki RESTful Application Programming Interface (API) [52] is available and it is mounted against the Meraki cloud platform. The Meraki API can

be used to interact and fetch data directly from the Meraki cloud. The API will perform a Hypertext Transfer Protocol Secure (HTTPS) request to the Meraki cloud, which will respond with the requested data in a JavaScript Object Notation (JSON) [53] format data set.

3.2.3 Speedtest.net

Speedtest.net is a web-based tool popular for testing the download and upload speeds of your internet connection [54].

3.2.4 iPerf3

iPerf3 is another tool to measure download and upload speed of a network. The main difference from Speedtest.net is that iPerf3 supports continuously measuring over longer periods of time, while Speedtest.net tests a small time window [55].

3.2.5 GPSLogger

GPSLogger is an Android application primarily used for logging GPS positions. It offers various configurations and supports automatic upload of logs to a selection of online storage solutions [56].

3.2.6 G-MoN Pro

G-MoN Pro is an Android application designed to display and log signal measurement data from the cellular network the UE is connected to [57].

3.2.7 Microsoft Teams

Microsoft Teams is a widely used video conferencing application [58].

3.2.8 PuTTY

PuTTY is an application that allows a computer to connect to other devices through a variety of interfaces [59].

3.2.9 Elastic Stack

An Elastic Stack consisted of two components: Elasticsearch and Kibana. It can be used to store, analyze, and visualize data [60].

Elasticsearch is a "JSON-based search and analytics engine" [61]. It can store data in a centralized location and also provides tools for searching through and analyzing the data.

Kibana is a component that connects to Elasticsearch and offers a web-based user interface for managing and visualizing data from Elasticsearch. Kibana provides tools for easily creating different graphs and maps from the data in Elasticsearch, offering various options for data analysis [60].

3.2.10 Python

Python is a high-level, object oriented programming language [62].

3.2.11 OBS Studio

OBS studio is screen recording software [63].

3.3 MG21 frequencies specification and radiation pattern

The following section will take a look at the MG21's radiation pattern and the frequencies used by the MG21, as well same how the RSRP and RSRQ values are interpreted by the MG21.

3.3.1 MG21 frequencies

The MG21 can operate on various frequencies to achieve connectivity to the 4G LTE network. The frequencies ranges that the MG21 can operate on are [64]:

- 829 MHz: frequencies range from 698 - 960 MHz
- 1940 MHz: frequencies range from 1710 - 2170 MHz
- 2495 MHz: frequencies range from 2300 - 2690 MHz

Even if the MG21 is able to operate on these frequencies, the frequency bands on the eNodeB in Norway must be compatible with the frequencies that the MG21 supports to access the 4G LTE network. The MG21 is also capable of receiving aggregated signals on the DL with carrier aggregation, where two component carriers can be used in the CA, achieving a bandwidth of 40 MHz. In Norway, the MG21 can utilize the 800 MHz + 1800 MHz frequency bands for CA [65].

3.3.2 internal Antennas and radiation pattern

The MG21 has two internal antennas: one main antenna and one auxiliary antenna. Auxiliary antennas in 4G LTE devices are used as secondary receiving antennas, serving the purpose of increasing connection quality and speed. It also serves the purpose as a backup antenna when the main antenna is not available [66].

The antenna type for both the main and auxiliary antennas is called a Planar Inverted-F Antennas (PIFA) [64]. PIFA is an omnidirectional antenna and is a very popular type used in many devices in the mobile phone market [67].

Figure 3.2 depicts the principal plane pattern for the main antenna of the MG21 at the frequency ranges of 829 MHz and 1940 MHz.

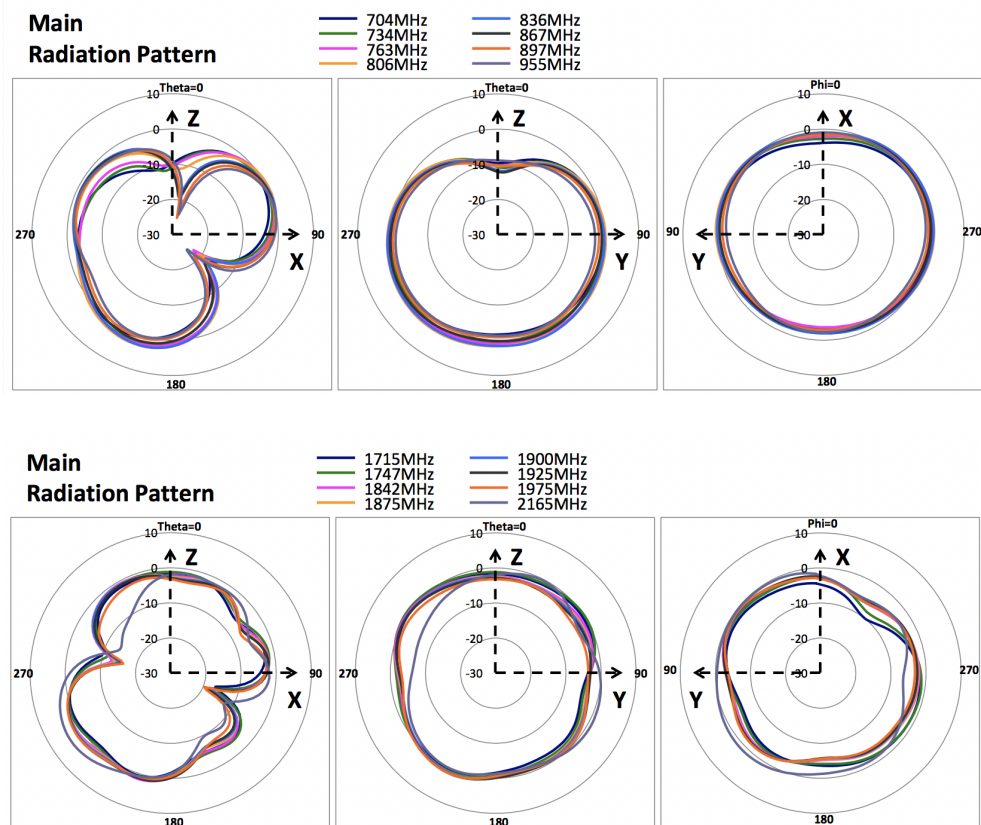


Figure 3.2: Radiation pattern for the MG21 for the 829 MHz and 1940 MHz frequency ranges.

Courtesy of Cisco Systems, Inc. Unauthorized use not permitted 15.05.2023 [64]

As shown in Figure 3.2, the main antenna experiences much more gain in the vertical plane (the xz-plane and yz-plane) than in the horizontal plane (the xy-plane) in the 829 MHz frequency range. Surprisingly, the radiation in the yz-plane on the 1940 MHz frequency range is better than in the xy-plane, which is not a characteristic of an omnidirectional antenna. A noticeable difference between the frequency ranges is that the 1940 MHz range has a higher peak gain than the 829 MHz range: 4.8 dBi for the main antenna and 3.8 dBi for the auxiliary antenna, compared to 2.9 dBi for the main antenna and 2.7 dBi for the auxiliary antenna [64].

The MG21 uses radio waves to communicate with the base station. There are several base stations from which the radio signals originate, either from very high places or from the ground. The omnidirectional radiation pattern of the MG21 is ideal for radio signals originating from the horizontal plane, where the signal travels horizontally and slightly upwards [35].

Having knowledge of the radiation pattern will provide the user with an idea of where to place the device to achieve the best possible connection to the eNodeB.

3.3.3 RSRP and RSRQ values

As mentioned in sections 2.3.1 and 2.3.2, both RSRP and RSRQ are parameters for measuring signal quality. Table 3.2 explains how Cisco Meraki interprets the different values of RSRP and RSRQ for their Meraki gateways in the 4G LTE network.

When RSRP under -140 dBm and RSRQ under -20 dB, it is considered as no signal reaching the UE. Cisco Meraki uses signal bars to measure the strength of a signal relative to the available signal power. A signal bar of five is considered as excellent signal strength, indicating that the UE is receiving the maximum data speed and capacity within the connected frequency band [68]:

Signal bar	RSRP(dBm)	RSRQ(dB)
5	RSRP \geq -83	RSRQ \geq -7
4	-83 > RSRP \geq -92	-7 > RSRQ \geq -10
3	-92 > RSRP \geq -102	-10 > RSRQ \geq -13
2	-102 > RSRP \geq -111	-13 > RSRQ \geq -16
1	-111 > RSRP \geq -140	-16 > RSRQ \geq -20

Table 3.2: Signal bar threshold for RSRP and RSRQ, specific for Cisco Meraki gateways [68].

3.4 Test area and signal readings from Telenor

For this project, Telenor has provided signal measurement data for the specific area where the tests will be executed. Telenor has previously conducted signal measurements for the test area assigned to the team using a software called TEST Mobile System (TEMS) [69]. TEMS has determined the eNodeB and frequency band that cover the test area, as well as measured the signal quality of the network. The TEMS readings were obtained using a device similar to a smartphone, specifically designed to measure network quality.

Telenor has provided data regarding the frequency band and eNodeB serving the test area, along with RSRP measurements. Direct access to the TEMS system was not granted; only relevant screenshots were provided.

3.4.1 Test Area

All of the tests will be conducted within the road section designated by Telenor, as mentioned earlier. The test area is located on Skonhovdvegen road in the 2827 Hunndalen postcode area, as depicted in Figure 3.3. Telenor chose this road section based on their previous signal readings from TEMS, which indicated inconsistent signal quality on Telenor's 4G network along this road. The purpose of selecting this area is to imitate the potential scenarios that may occur at the quarries.

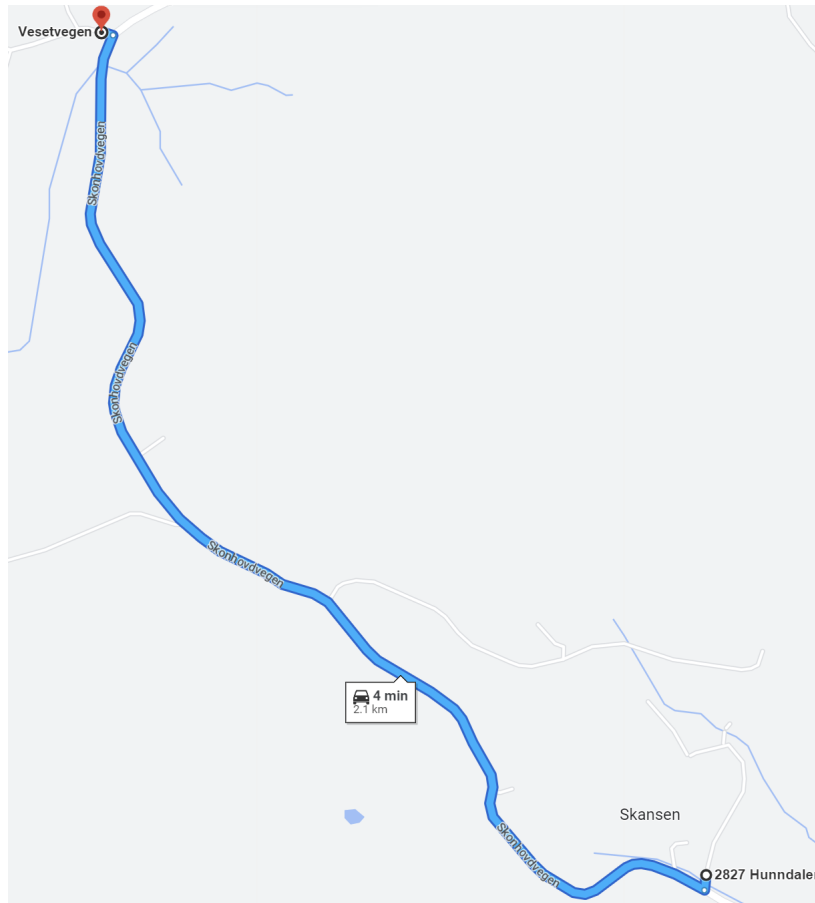


Figure 3.3: Test area used throughout all the tests. Taken from Google Maps

3.4.2 Best coverage eNodeB

There are primarily three different eNodeBs that cover our test area to provide access for UEs to Telenor's 4G LTE network. Figure 3.4, taken from the TEMS reading, shows the locations of the eNodeBs: Skumsjovegen (A), Lauvkampen (B) and Vardalsaasen (C), which covers the given test area.

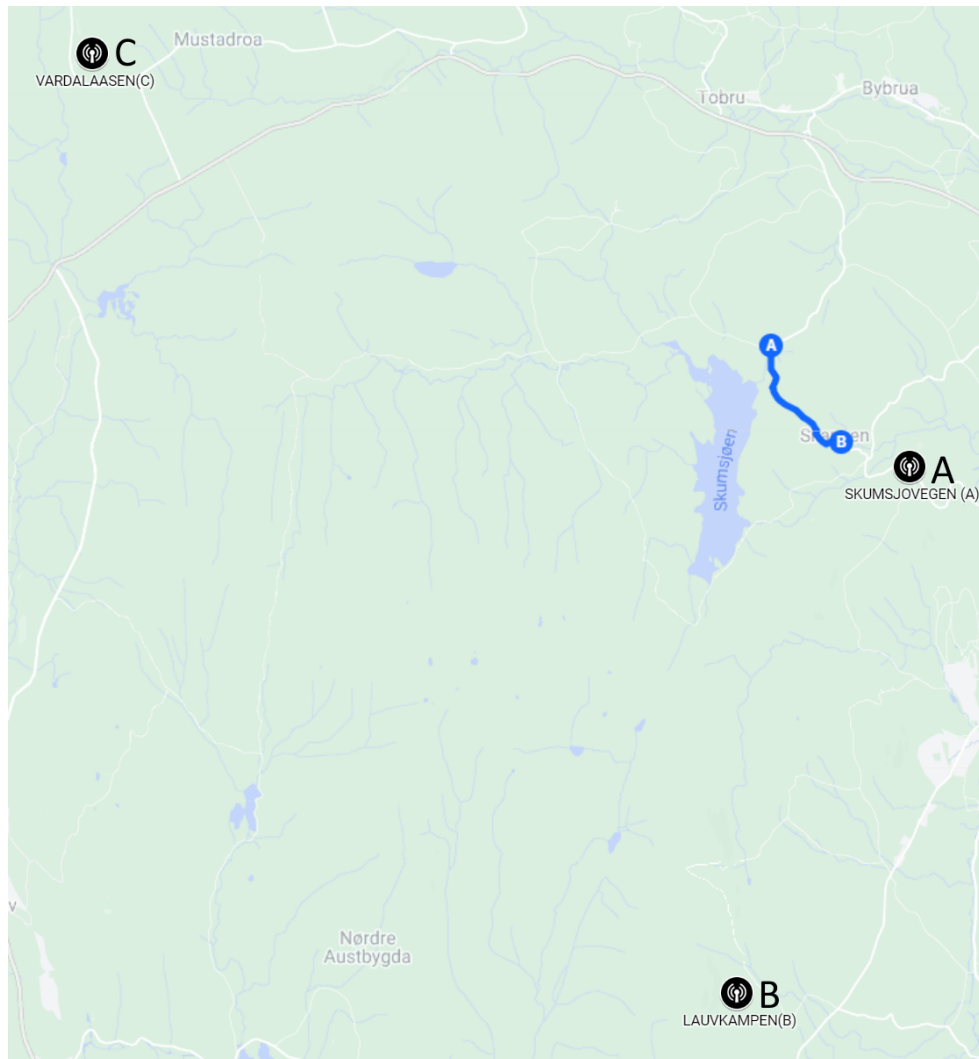


Figure 3.4: The placement of the eNodeBs relative to the test area. Skumsjovegen (A), Lauvkampen (B), Vardalsaasen (C). Taken from Google Maps

3.4.3 800 MHz frequency band (B20) coverage

Figure 3.5 illustrates the 800 MHz frequency band coverage from Skumsjovegen (yellow), Vardalsaasen (orange), and Lauvkampen (red) in the test area captured using TEMS.

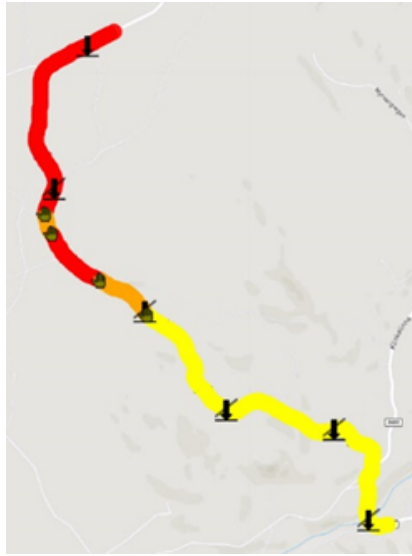


Figure 3.5: Illustration of which eNodeB is providing 800 MHz in each section of the test area. Lauvkampen (red), Vardalsaasen (orange), Skumsjovegen (yellow), taken from TEMS

Coverage of the 800 MHz frequency band seems to be available throughout the test area. In theory, a UE should be able to establish a connection to the 4G LTE network anywhere in the test area, assuming no significant signal interference.

Mainly, the connection will be established between the eNodeB on Lauvkampen (red) in the northwest section of test area, while the southeast section of the test area is mainly covered by the eNodeB on Skumsjovegen (yellow). There are two small sections across the test area where a cell handover is performed with the eNodeB found on Vardalsaasen (orange).

3.4.4 1800 MHz frequency band (B3) coverage

The eNodeB on Skumsjovegen can provide the UE with the possibility to connect to the 1800 MHz frequency band, which offers extended capacity and speed for the 4G network. The red dots in Figure 3.6 illustrates the coverage of the 1800 MHz frequency band, while the orange dots represent the coverage of the 800 MHz frequency band, also from Skumsjovegen.

Skumsjovegen is geographically close to the test area, and the 1800 MHz frequency band is a capacity band, which means it won't cover as large of an area as the 800 MHz frequency band. The other eNodeBs on the other locations may also have the 1800 MHz frequency band installed, but their coverage isn't large enough to reach the test area.

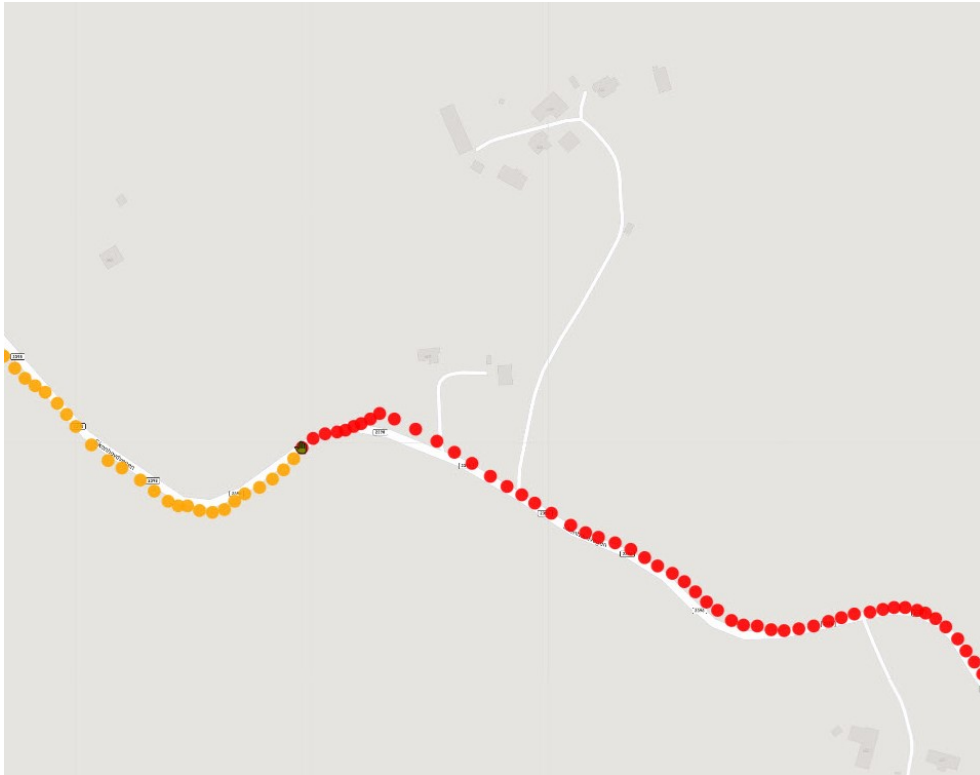


Figure 3.6: Illustration shows the southernmost part of the test area (Figure 3.5) where there is 1800 MHz coverage from Skumsjovegen: the red area represents the 1800 MHz frequency band, and the orange area represents the 800 MHz frequency band. Taken from TEMS

3.4.5 RSRP readings

Telenor has also provided detailed RSRP readings from the eNodeB in Skumsjovegen. The RSRP readings for the 1800 MHz frequency band, represented from bottom right to left right before the handover (indicated by the star symbol) to the 800 MHz frequency band, can be observed in Figure 3.7. The corresponding color-coding of RSRP values shown in Figure 3.8 explains the significance of each color. Starting from the point in the bottom right, the RSRP is relatively strong, but as one moves further up the road and away from the eNodeB on Skumsjovegen, the signal strength weakens.

Right before the handover, the signal strength is at its weakest and almost non-existent in the 1800 MHz frequency band. The TEMS-utilized UE decides to switch to the frequency band with higher coverage to obtain a stronger signal. The handover will occur within the same eNodeB and transition to the 800 MHz coverage band.

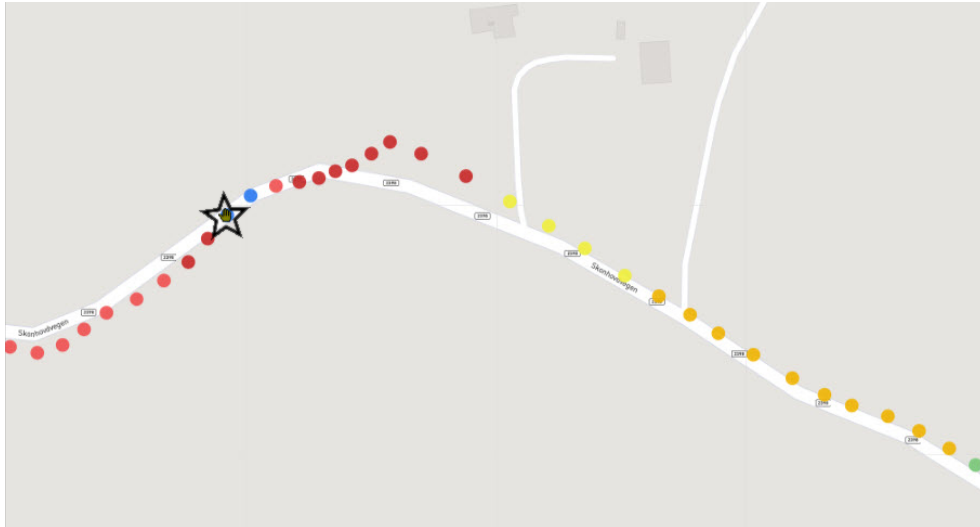


Figure 3.7: RSRP values for the same section of the test area shown in Figure 3.6. The right star symbol represents RSRP for 1800 MHz, while the left star symbol represents RSRP for 800 MHz. Taken from TEMS

Color	RSRP values
Dark Green	$-80 \leq \text{RSRP} \leq 25$
Green	$-85 \leq \text{RSRP} \leq -80$
Light Green	$-90 \leq \text{RSRP} \leq -85$
Yellow	$-95 \leq \text{RSRP} \leq -90$
Orange	$-100 \leq \text{RSRP} \leq -95$
Red	$-105 \leq \text{RSRP} \leq -100$
Dark Red	$-110 \leq \text{RSRP} \leq -105$
Blue	$-115 \leq \text{RSRP} \leq -110$
Light Blue	$-120 \leq \text{RSRP} \leq -115$
Purple	$-160 \leq \text{RSRP} \leq -120$

Figure 3.8: Color codes for TEMS reading related to figure 3.7

3.5 Setup

This section will go deeper into how the hardware and the software were set up for the tests, specifically how the hardware was set up in the car, how the components connect to each other, and how the software interacts with them. This section will also take a look at some of the problems that were discovered during the project and the differences between the original plan and the final setup.

3.5.1 Initially planned setup

The initially planned test setup would consist of the MG21, the Thingy 91, and a laptop, as illustrated in Figure 3.9.

The MG21 would be connected to the laptop via an Ethernet cable to give the laptop access to Telenor's 4G network. The MG21 would also measure the signal quality of Telenor's 4G network.

The Thingy 91 would be connected to the laptop via a USB cable to log the output. It would also be able to connect to Telenor's 4G network to measure the signal quality of the network. Additionally, it would gather GPS positioning to know exactly where the signal measurement was taken throughout the test area.

As seen in Figure 3.10, the MG21 was placed on the car dashboard on the passenger side and secured with duct tape, while the Thingy 91 was placed on the dashboard and secured with Velcro.

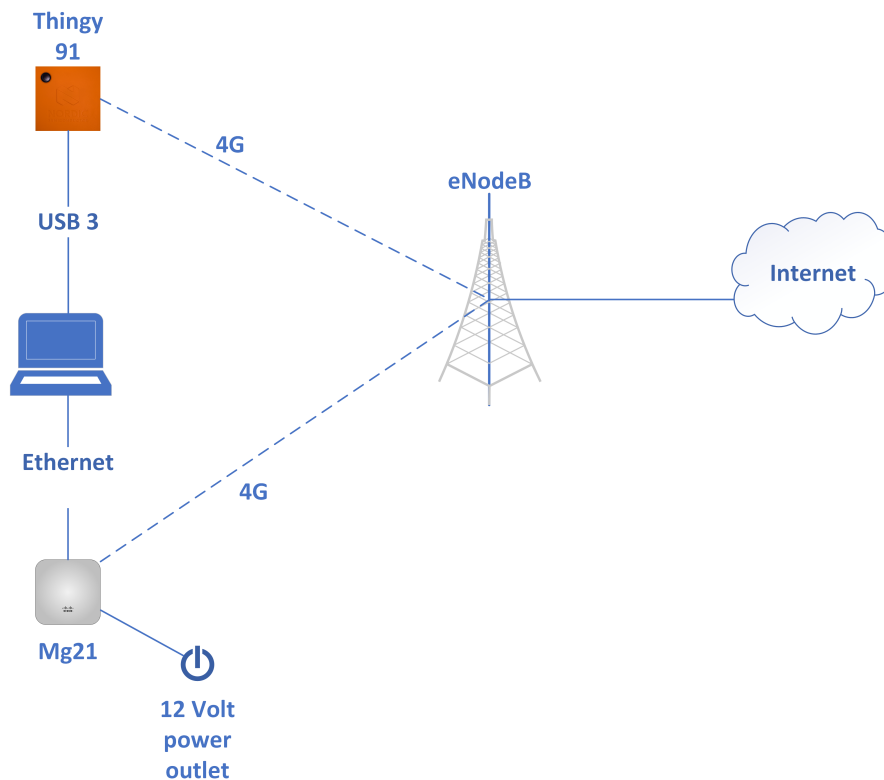


Figure 3.9: Illustration of the initial planned test setup.



Figure 3.10: How the hardware was initially set up in the car.

The software used during the initially planned setup included Microsoft Teams for the live conference call for data traffic, PuTTY to log the Thingy 91 positional data, a Python script to collect data from the Meraki API, OBS Studio for screen capture, and a Speedtest.net window in a web browser to periodically measure the current bandwidth.

3.5.2 Changes to setup

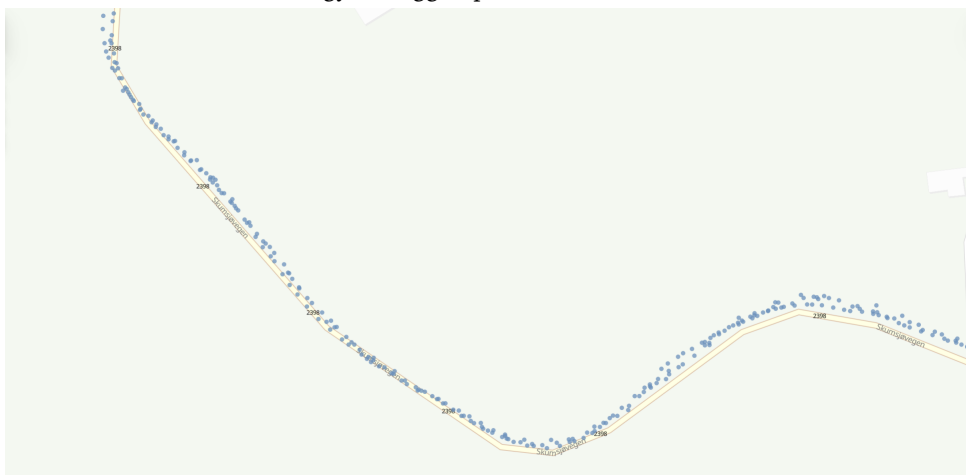
There was a change to the hardware setup after a pilot test was performed to check if everything worked as desired. The Thingy 91 was replaced by a smartphone. The main reason for this hardware change was the lack of precision in the Global Positioning System (GPS) positional data from the Thingy 91. Figure 3.11a shows some of the GPS data points to be way off the road section that was driven, and they were very inconsistent.

During the pilot test a smartphone running the GPSLogger application was tested to determine the accuracy of the GPS positioning results and compare it to the Thingy 91. After testing, Figure 3.11b confirmed that the smartphone was

more accurate than the Thingy 91 with GPS positioning. As a result, it was decided to replace the Thingy 91 with the smartphone mentioned in Section 3.1.3 for the remainder of the testing phase.



(a) Thingy 91 logged positions for the first test



(b) GPSLogger logged positions for the first test

Figure 3.11: Comparison between the logged positions from the Thingy 91 and smartphone using GPSLogger

Additionally, a couple of changes were made to the software for the final setup. The smartphone had GPSLogger added for GPS positioning, as mentioned earlier. Since the Thingy 91 also received signal measurement data from the 4G network, the smartphone also ran the G-MoN application to collect signal data from Telenor's 4G network.

Furthermore, some software changes were made with the laptop connected to the MG21. A web browser window displaying a live clock was added to provide a clearly visible time on the recorded videos. Additionally, a web browser window with the Meraki dashboard was included to monitor the current frequency band to which the MG21 was connected. This addition was made because the Meraki API does not gather information about the frequency bands used, and this data might be useful for analysis when discussing the superparameter.

The web browser window with Speedtest.net was replaced with a console window for iPerf3. Although the console window couldn't display data live while logging, it was used to ensure that iPerf3 was running correctly. The plan was also to use iPerf3 to measure the upload and download speeds of both the smartphone and the MG21. However, several tests revealed issues with iPerf3 when running it on the smartphone. The measurements were inconsistent, resulting in extremely unstable results even when the smartphone remained stationary with consistent signal parameters. Consequently, it was decided to exclude iPerf3 readings from the smartphone.

An updated version of the Python script was added to collect data from the Meraki API. One of the main additions was the ability to set the direction in which the car was driving through the Graphical user interface (GUI), which enabled us to compare data from the different driving directions.

3.5.3 Final setup

The final hardware setup consisted of the MG21, a smartphone, a laptop, an external laptop, and an external server. Figure 3.12 provides a full overview of all the components that were used during the tests.

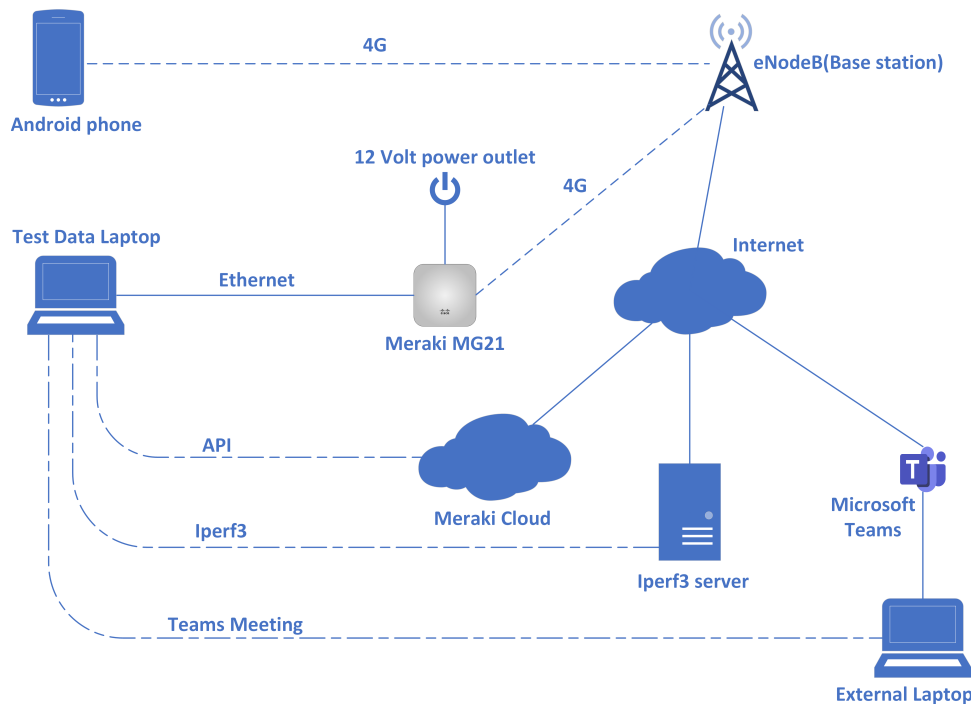


Figure 3.12: Technical outline of the final test setup.

The MG21 and the smartphone were used for 4G connectivity and signal measuring of Telenor’s 4G network. Both devices used 4G to connect to eNodeBs, which provided the smartphone and MG21 with internet access.

A laptop was connected to the MG21 through an Ethernet cable, allowing it to access the 4G network. This enabled the Python script to run on the laptop to gather signal data from the Meraki cloud, using the Meraki API. The laptop also opened a web browser window to access the Meraki dashboard to gather the information about the frequency bands in use, which was not possible to access via the Meraki API.

The laptop served also as a client to perform upload and download speed tests using iPerf3. An external server was used as an iPerf3 server.

Additionally, the laptop used the internet connection to participate in a Microsoft Teams conference meeting with another external laptop. This was used to maintain a continuous traffic flow.

All the activities on the laptop were recorded via OBS Studio. The recordings were used to manually insert the frequency band data by watching the recordings and inserting the data in a log file.

The smartphone ran GPSLogger to log GPS positioning and used G-MoN to collect signal measurement data. G-MoN can log similar signal measurement data as the Meraki API, but at more frequent intervals. One notable difference is that G-MoN can also log the frequency band and if CA was used by the smartphone, as well as the specific eNodeB with which the UE is associated, unlike the Meraki API.

In Figure 3.13, the MG21 is attached to the side window of the car using duct tape, while the smartphone is placed on the car's dashboard. This placement was chosen to provide the best possible signal conditions for the devices, ensuring that the antennas were obstructed by as few objects as possible to avoid interference that could affect the signal quality.



Figure 3.13: The final hardware setup inside the car.

3.6 Test execution

The tests are conducted by setting up the hardware as described in Section 3.5. The car is then driven through the test area at a very low speed, approximately 5 km/h.

An inconvenience discovered with the Meraki API was that the data fetched from the API isn't gathered as frequently as the data on the Meraki dashboard. There is an approximately 45-second delay between each request to the Meraki cloud, meaning that the API calls only retrieve new data every 45 seconds. This was the reason for the low driving speed. Additionally, the API is unable to fetch all the available data from the Meraki cloud, such as the frequency band to which the MG21 is connected.

All of the data-gathering equipment collects data periodically on set intervals. The one second interval for the other software were chosen to easily match them with the data from the MG21. Driving at this slow speed ensures enough data points from the MG21 and helps distribute them evenly across the test area.

A test day will usually consist of three "tests", where one test is defined as driving back and forward in the test area three times. Driving back and forth once is defined as a "round". The reason behind executing so many test, is to capture enough sufficient data to analyze the result thoroughly and have enough data to discuss a possible superparameter propositions. Due to the low speed, a single test takes approximately two hours of continuous driving.

The test will be preformed at different times of the day, one test in the morning, one at mid-day, and the last test in the evening. The tests are performed at these times, mainly to assess if network congestion (Section 2.1.5) and different weather conditions may effect the network and signal quality of the 4G network.

3.7 Data processing

During the tests, several devices collected data to multiple log files. To begin analyzing the data, it needs to be processed into a more manageable format, as shown in Figure 3.14. During the processing, the data goes through several steps, most of which are handled by a Python script.

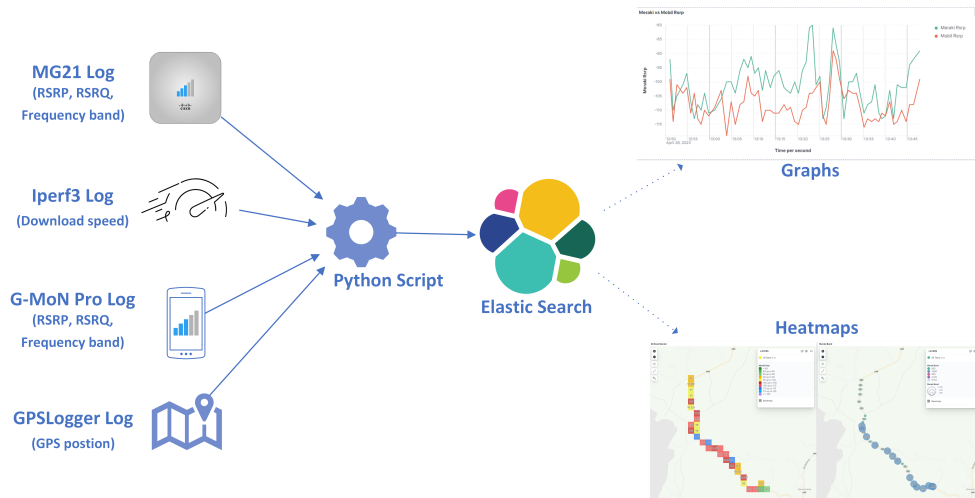


Figure 3.14: Illustration of the data processing process

The first step of this process involves adding data that is not automatically logged during testing to a log file. This is the most manual step in the process and mainly involves the frequency band used by the MG21, which is not accessible from the Meraki API. The data was obtained from the Meraki dashboard using recordings from the tests, and a Python script was used to add it to the log file for the MG21.

The second step involves taking all the data from the different log files and preparing them for Elasticsearch. During this step, all log files are converted to Comma-separated values (CSV) files, and additional fields are added based on existing fields to fit with different Elasticsearch functions. For example, latitude and longitude are combined into a single location field that Kibana can use as a Geo point on a map. It also includes creating secondary fields for values like RSRP, which is normally a negative value, where it is written as a positive value. This was needed as some of the functions used in the maps in Kibana was not compatible with negative values at the time of the project.

The final step of the process is uploading the data to Elasticsearch for analysis. A dashboard is created in Kibana with various visualizations to assist in data analysis. The Python script that processes the data handles the data upload and also creates a single CSV file containing all the data as a local backup. This entire process is repeated twice for each test: once aligning everything with the data from the MG21 and once using only the data from the smartphone. This is done because the smartphone collects a greater number of data points, providing a more detailed perspective on the tests.

3.8 The playbook

One of the objectives of the project was to create a playbook, the content of this playbook is based on Telenor's requirements. This playbook can be found in Appendix F. It is made up of a presentation, a text document and a demo video. The purpose of the presentation is to introduce the project to individuals who are not IT professionals. It is designed to be easily understandable and provide a general overview of how the data was collected and processed. The presentation consists of one slide explaining the execution of tests and another slide focusing on data processing. The text document serves as a step-by-step guide on navigating the Kibana, aiding someone in delivering a demonstration of the visualized heatmaps and graphs within Kibana. The video serves as a backup solution for the Kibana demonstration in case of technical difficulties, covering most of the content that would have been shown during a live demonstration.

3.9 Hypotheses

The hypotheses in this section are based on expectations for what the results from the tests might be. These expectations are related to the theory from Chapter 2 and from the data from Telenor's TEMS measurements.

3.9.1 Result compared with Telenor's previous TEMS data

The hypothesis will heavily rely on Telenor's signal tests done with TEMS in the specific test area, as mentioned in Section 3.4. Throughout the majority of the test area, the UEs will connect to the 800 MHz frequency band and switch to the 1800 MHz frequency band when they approach the section illustrated in Figure 3.6, where the 1800 MHz frequency band is available. Although the TEMS device did not detect any other locations within the test area where it could connect to the 1800 MHz frequency band, the MG21 might connect to the 1800 MHz frequency band in other places throughout the test area. Even if the TEMS device did not identify any other available areas with coverage from the 1800 MHz frequency band, it is expected that the MG21 might be able to connect 1800 MHz from another eNodeB within the test area.

The MG21 most likely has a stronger antenna and better antenna radiation than a regular smartphone and the TEMS devices, due to the fact that the antennas on the MG21 are bigger and there is an auxiliary antenna available. This enables the MG21 to pick up signals from eNodeBs that are further away. It may even encounter frequencies from other frequency bands besides the 800 MHz and 1800 MHz frequency bands. On the other hand, the smartphone is expected to be connected to the same frequency bands as indicated by Telenor's TEMS readings since the antennas used on the TEMS devices likely have similar specifications to those found in modern smartphones, as the TEMS devices used by Telenor are mostly modified smartphones.

- **Hypothesis 1 (H1):** *The MG21 has the capability to establish connections to the 1800 MHz frequency bands at multiple locations within the test area.*
- **Hypothesis 2 (H2):** *The MG21 is able to establish a connection to frequency bands other than the 800 MHz and 1800 MHz frequency bands.*
- **Hypothesis 3 (H3):** *The smartphone will establish a connection to only the 800 MHz and 1800 MHz frequency bands.*

3.9.2 Cell selection and handover

Telenor's TEMS readings shows places within the test area where the TEMS device switches back and forth between the same eNodeBs. A drop in signal quality and overall user experience may be expected during the handover process due to the delay associated with the handover interruption time mentioned in Section 2.4.3. While the discrepancy is not expected to last for long periods of time and it might have an unnoticeable effect on the user experience, it is expected to be seen in the iPerf3 logs that the bandwidth speed drops during handovers.

- **Hypothesis 4 (H4):** *During handovers, the signal quality and the bandwidth will decrease for a short duration.*

3.9.3 Driving direction

It is expected that the driving direction of the vehicle will have a minimal effect on the quality of the 4G network for the smartphone. The smartphone is expected to switch between different cells with minimal changes in network quality. The eNodeBs that cover the test area have long distances between each other, leading to a low probability of a cell having better signal quality on one side of the road than the other. Therefore, it is expected that the smartphone will connect to the same frequency bands when in the same location within the test area, regardless of

the direction the vehicle is headed. However, the smartphone is expected to have slightly different values for RSRP and RSRQ depending on the driving direction, as the vehicle's rotation may cause the radiation pattern from the smartphone to favor one direction slightly over the other.

The MG21, being a stationary device, might be more affected by the driving direction. It is not accustomed to moving between different cells quickly, and therefore, the device might attempt to stay connected to a cell from an eNodeB as long as possible. Consequently, it is predicted that the MG21 will connect to a different frequency band within the same section of the test area, depending on the way the vehicle is headed. The values of RSRP and RSRQ will also be affected since they would need to be very low before the MG21 decides to perform a handover.

- **Hypothesis 5 (H5):** *The driving direction will have minimal effect on the 4G network and signal quality, as well as the frequency band usage for the smartphone.*
- **Hypothesis 6 (H6):** *The driving direction will have a noticeable effect on the 4G network and signal quality, as well as the frequency band usage for the MG21.*

3.9.4 Bandwidth

The prediction is that the bandwidth is reduced when the UE is connected to a frequency band with a lower frequency, as mentioned in Section 2.1.6. A higher frequency will result in better bandwidth and capacity for the UE. Therefore, it is expected that the bandwidth and the QoS will always be better in the 1800 MHz frequency band compared to the 800 MHz frequency band.

The amount of traffic present in the network is expected to impact the capacity and the bandwidth of the tested 4G network as mentioned in Section 2.1.5. There is no way to determine how crowded Telenor's 4G network is during the tests, but one can assume that the traffic will be highest during tests conducted in the mid-day. This assumption is based on the fact that most people finish work around this time, and they are likely to utilize the cellular network while driving home.

- **Hypothesis 7 (H7):** *The 1800 MHz frequency band will always achieve a higher bandwidth than the 800 MHz frequency band*
- **Hypothesis 8 (H8):** *The amount of users present on the 4G network will affect the bandwidth and the capacity.*

3.9.5 RSRP and RSRQ data

It is expected that there is a correlation between a drop in the signal parameter RSRP and the initiation of a switch to another frequency band. It is anticipated that when the RSRP value gradually decreases, the UEs will attempt to change to a frequency band with a lower frequency to improve coverage. Similarly, when the RSRP value gradually increases, the UEs will perform a switch to a frequency band with a higher frequency to get better bandwidth.

After any frequency switch, it is expected that the UEs will either maintain or experience increased values of RSRP. That's because when the UEs decide which cell to switch over to, a cell selection process is performed primarily based on the RSRP, where a better value than the current serving cell is desired (more about this in Section 2.3.1). Therefore, it is expected that the neighboring cell will always have a better or the same RSRP as the serving cell.

Both devices are expected to exhibit similar trends in RSRP and RSRQ values. When the RSRP decreases, the RSRQ will also decrease, although not at the same rate. This difference is due to the fact that RSRQ relies not only on RSRP but also on interference that affects the radio signal. However, higher RSRP and RSRQ values are generally expected on the MG21 compared to the smartphone, due to the expected superior antenna radiation of the MG21, providing a stronger signal across different cells.

An exception arises where the 800 MHz frequency band is expected to yield better RSRP and RSRQ value than the 1800 MHz frequency band. This is based on the theory in Section 2.1.6, that the coverage bands offer improved coverage and superior penetration through physical objects compared to capacity bands.

- **Hypothesis 9 (H9):** *When the RSRP gradually decrease, the UEs will change to a lower frequency band. When the signal parameters gradually increase the UEs will change to a higher frequency band.*
- **Hypothesis 10 (H10):** *After a frequency switch, the UEs will maintain or have an increased RSRP.*
- **Hypothesis 11 (H11):** *RSRP and RSRQ will follow the same trends throughout all the tests.*
- **Hypothesis 12 (H12):** *The MG21 will have higher RSRP and RSRQ than the smartphone.*

- **Hypothesis 13 (H13):** *There will be an overall lower RSRP and RSRQ on the 1800 MHz frequency band than on the 800 MHz frequency bands.*

3.9.6 Weather

The hypothesis is that the weather conditions will affect the quality of the signal during a test. For example heavy rain or snow will somewhat obstruct the path from the eNodeB to the UE, affecting the signal quality (Section 2.1.4). Very thick fog will have a similar effect on the quality of the signal. Furthermore differences in temperature will not have any noticeable effect.

- **Hypothesis 14 (H14):** *The weather condition will have an effect on the 4G signal quality.*

3.10 Method summary

This section contains a summary of this chapter, focusing on test execution, test setup, as well as data processing.

3.10.1 Final setup

The final hardware setup consisted of the MG21, a smartphone, a laptop, an external laptop, and an external server. The software used consisted of GPSLogger and G-MoN used on the smartphone and a Python script, the Meraki dashboard, iPerf3 client, OBS Studio, Microsoft Teams and a live clock on a laptop. An external laptop was used to host the Microsoft Teams meeting, and an external server was set up as the iPerf3 server. A technical outline of the setup can be seen in Figure 3.15.

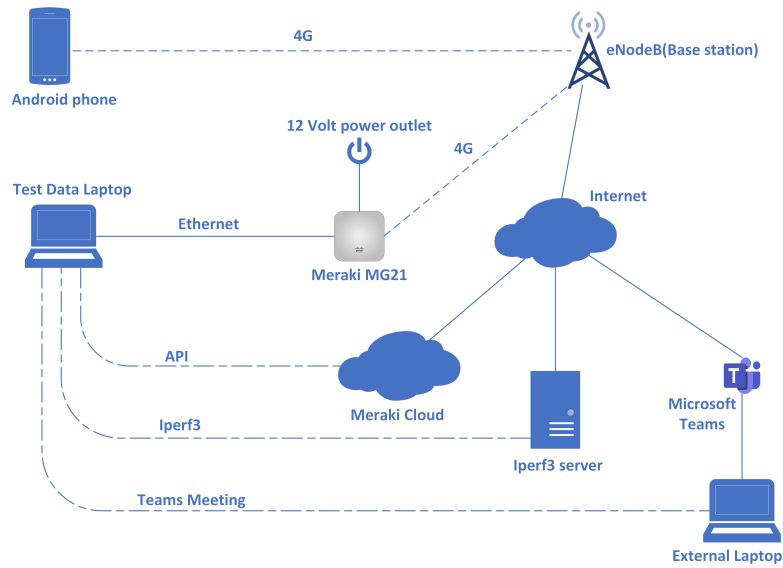


Figure 3.15: Technical outline of the final test setup. (Reused Figure 3.12)

3.10.2 Test execution

Tests were conducted in a designated test area at different times of the day. One test consists of driving back and forth between point A and B, as shown in figure 3.16, for three rounds. Each round represents one complete trip from point A to B and back. The car maintained a speed of 5 km/h due to the Meraki API update delay. Efforts have been made to conduct separate tests on a single day, whenever possible, in order to observe any noticeable differences based on the time of day.

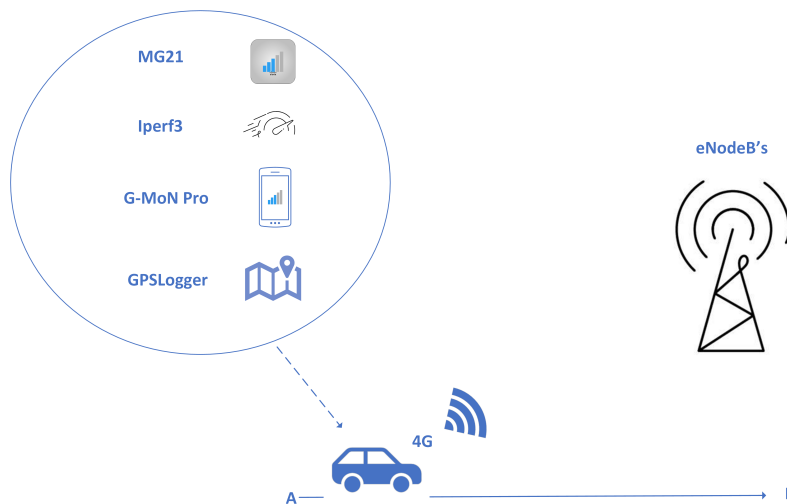


Figure 3.16: Simplified figure of how the tests were performed. (Reused Figure 3.1)

3.10.3 Data processing

The data processing consisted of gathering all the data from the laptop and the smartphone from a test, manually adding the frequency bands from the screen recording, using scripts to convert all the logs to a CSV format. Then data required for Kibana was generated, and finally all the logs was merged into combined logs using timestamps as the primary reference point. Finally the script uploaded the combined logs to Elasticsearch, which were then visualized in a Kibana dashboard. An illustration of the data processing performed by the scripts can be seen in figure 3.17.

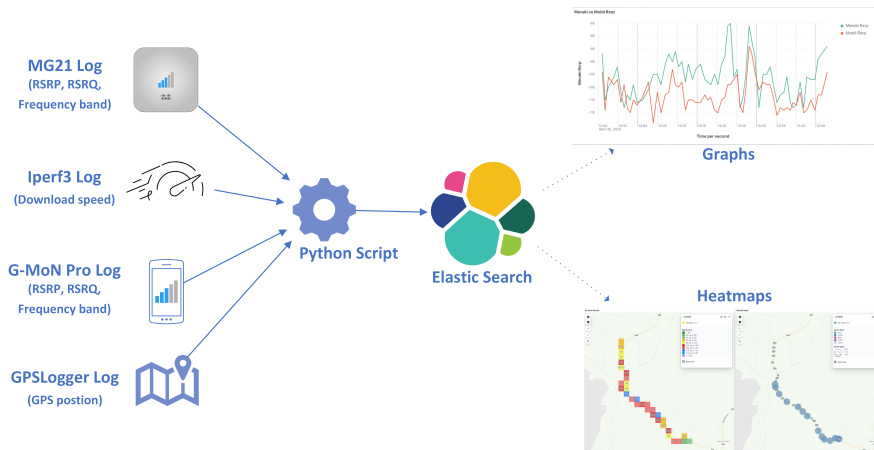


Figure 3.17: Illustration of the data processing process (Reused Figure 3.14)

3.10.4 Hypotheses

This subsection includes a condensed list of all the hypotheses.

- **Hypothesis 1 H1:** *The MG21 has the capability to establish connections to the 1800 MHz frequency bands at multiple locations within the test area.*
- **Hypothesis 2 H2:** *The MG21 is able to establish a connection to frequency bands other than the 800 MHz and 1800 MHz frequency bands.*
- **Hypothesis 3 H3:** *The smartphone will establish a connection to only the 800 MHz and 1800 MHz frequency bands.*
- **Hypothesis 4 H4:** *During handovers, the signal quality and the bandwidth will decrease for a short duration.*
- **Hypothesis 5 H5:** *The driving direction will have minimal effect on the 4G network and signal quality, as well as the frequency band usage for the smartphone.*

- **Hypothesis 6 H6:** *The driving direction will have a noticeable effect on the 4G network and signal quality, as well as the frequency band usage for the MG21.*
- **Hypothesis 7 H7:** *The 1800 MHz frequency band will always achieve a higher bandwidth than the 800 MHz frequency band*
- **Hypothesis 8 H8:** *The amount of users present on the 4G network will affect the bandwidth and the capacity.*
- **Hypothesis 9 H9:** *When the RSRP gradually decrease, the UEs will change to a lower frequency band. When the signal parameters gradually increase the UEs will change to a higher frequency band.*
- **Hypothesis 10 H10:** *After a frequency switch, the UEs will maintain or have an increased RSRP.*
- **Hypothesis 11 H11:** *RSRP and RSRQ will follow the same trends throughout all the tests.*
- **Hypothesis 12 H12:** *The MG21 will have higher RSRP and RSRQ than the smartphone.*
- **Hypothesis 13 H13:** *There will be an overall lower RSRP and RSRQ on the 1800 MHz frequency band than on the 800 MHz frequency bands.*
- **Hypothesis 14 H14:** *The weather condition will have an effect on the 4G signal quality.*

Chapter 4

Results and Discussion

This chapter will examine the results from the frequency bands, RSRP, RSRQ, and upload/download speed through the MG21. frequency bands, RSRP, and RSRQ are also collected through the smartphone's 4G connection to Telenor's cellular network using G-MoN. G-MoN also shows us which eNodeB the smartphone is connected to and whether or not it is using CA at all times. Heatmaps will be used to display RSRP and RSRQ data taken from Elasticsearch, along with different types of graphs and tables to present the results in an understandable manner.

All logs from the test with the data used in this project can be found in the GitHub repository found in Appendix C.

NOTE: It's important to note that both RSRP and RSRQ are measured with negative values. However, the heatmaps displayed in this chapter will show RSRP and RSRQ with positive numbers. This is due to a limitation in Kibana when using negative values in heatmaps (further explanation in Section 3.7).

4.1 Showcasing overall result

Table 4.1 shows the overall results from all the tests conducted with the final setup (Section 3.5.3). Throughout Chapter 4, Table 4.1 will be frequently referred to in order to showcase the results from each of the tests.

After the last planned testing on 17.03, it was discovered that iPerf3, by default, measured upload speed, which was not ideal. This is because download speed is probably more commonly used in the use case for downloading data to calculate positions more accurately and to receive instructions. Therefore, two new tests were conducted on 26.04. The goal of these tests was to gather data on the download speed and determine if it behaved differently than the upload speed.

In total, there were four test days, with three tests performed on 13.03 and 17.03. The test conducted on 01.03 had only one test, while two test were performed on the 26.04. The first seven tests measured upload speed, while the last two tests measured download speed.

Dates	01.03.23		13.03.23	
Test nr.	Test 1	Test 1	Test 2	Test 3
Time	14:04 - 16:12	8:38 - 10:42	13:49 - 15:52	19:32 - 21:42
MG21 entries	146	151	140	146
MG21 max RSRP	-73 dBm	-71 dBm	-78 dBm	-78 dBm
MG21 min RSRP	-114 dBm	-114 dBm	-114 dBm	-116 dBm
MG21 avg RSRP	-99.08 dBm	-99.31 dBm	-98.76 dBm	-98.53 dBm
MG21 max RSRQ	-7 dB	-7 dB	-7 dB	-7 dB
MG21 min RSRQ	-14 dB	-14 dB	-13 dB	-14 dB
MG21 avg RSRQ	-10.07 dB	-9.71 dB	-9.62 dB	-9.9 dB
G-MoN entries	4351	4555	3440	2329
G-MoN max RSRP	-82 dBm	-81 dBm	-80 dBm	-82 dBm
G-MoN min RSRP	-120 dBm	-120 dBm	-122 dBm	-120 dBm
G-MoN avg RSRP	-103.6 dBm	-103.94 dBm	-106.38 dBm	-103.46 dBm
G-MoN max RSRQ	-4 dB	-4 dB	-4 dB	-5 dB
G-MoN min RSRQ	-16 dB	-16 dB	-16 dB	-16 dB
G-MoN avg RSRQ	-8.5 dB	-8.05 dB	-8.57 dB	-8.51 dB
iPerf3 entries	7025	6913	6701	7519
Max speed	46.18 Mbps	47.58 Mbps	45.89 Mbps	48.13 Mbps
Min speed	0 Mbps	0 Mbps	0 Mbps	0 Mbps
Avg speed	12.14 Mbps	13.73 Mbps	13.39 Mbps	15.9 Mbps

(a) Statistics showcasing the result from the smartphone and the MG21 from the test days 01.03 and 13.03.

Dates	17.03.23			26.04.23	
Test nr.	Test 1	Test 2	Test 3	Test 1	Test 2
Time	07:50 - 10:07	12:54 - 14:22	17:09 - 19:01	07:44 - 09:35	11:48 - 13:42
MG21 entries	135	120	142	128	110
MG21 max RSRP	-75 dBm	-80 dBm	-76 dBm	-72 dBm	-80 dBm
MG21 min RSRP	-116 dBm	-115 dBm	-115 dBm	-114 dBm	-113 dBm
MG21 avg RSRP	-98.27 dBm	-99.95 dBm	-101.35 dBm	-99.63 dBm	-99.7 dBm
MG21 max RSRQ	-7 dB	-7 dB	-7 dB	-9 dB	-9 dB
MG21 min RSRQ	-14 dB	-15 dB	-17 dB	-17 dB	-17 dB
MG21 avg RSRQ	-9.6 dB	-10.02 dB	-10.35 dB	-13.29 dB	-12.96 dB
G-MoN entries	3456	3727	3753	3945	3188
G-MoN max RSRP	-83 dBm	-86 dBm	-86 dBm	-83 dBm	-88 dBm
G-MoN min RSRP	-122 dBm	-123 dBm	-122 dBm	-119 dBm	-120 dBm
G-MoN avg RSRP	-108.47 dBm	-107.65 dBm	-108.12 dBm	-106.41 dBm	-108.02 dBm
G-MoN max RSRQ	-5 dB	-5 dB	-5 dB	-5 dB	-6 dB
G-MoN min RSRQ	-16 dB	-17 dB	-17 dB	-18 dB	-19 dB
G-MoN avg RSRQ	-8.72 dB	-8.89 dB	-9.09 dB	-10.54 dB	-11.38 dB
iPerf3 entries	6928	6490	6723	6734	7161
Max speed	51.41 Mbps	46.14 Mbps	51.61 Mbps	168.12 Mbps	164.37 Mbps
Min speed	0 Mbps	0 Mbps	0 Mbps	0.5 Mbps	1.19 Mbps
Avg speed	13.65 Mbps	11.49 Mbps	11.22 Mbps	43.92 Mbps	37.85 Mbps

(b) Statistics showcasing the result from the smartphone and the MG21 from the test days 17.03 and 26.04.

NOTE: Speed from iPerf3 measured the download speed for the tests performed 26.04

Table 4.1: Tables containing general statistics

4.2 Frequency Bands

This section will focus on the frequency bands used by the MG21 and the smartphone during the tests. It will examine which frequency bands they utilized the most, and whether they were connected to the expected frequency bands at different points within the test area.

4.2.1 Frequency bands utilized

Both the MG21 and the smartphone connected to the 1800 MHz and 800 MHz frequency bands for the majority of the test area, as shown in Figure 4.1. However, the 1800 MHz and 800 MHz frequency bands were not the only frequency bands to which the test devices connected throughout the tests. The TEMS results only identified the 1800 MHz and 800 MHz frequency bands (as seen in Sections 3.4.3 and 3.4.4). Additionally, the test results revealed that the 900 MHz and 2100 MHz frequency bands were used a few times by both devices. The MG21 utilized these frequency bands more frequently than the smartphone.

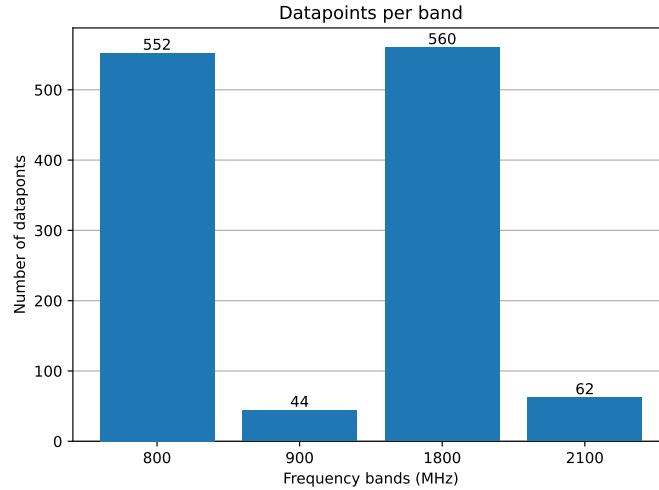
The 900 MHz (3GPP B8 frequency band) is also a coverage band just like the 800 MHz frequency band, but with slightly better capacity and lower coverage than the 800 MHz frequency band [70]. The 2100 MHz (3GPP B1 frequency band) is also a capacity band that is providing a better capacity than 1800 MHz frequency band, but with a smaller coverage area than the 1800 MHz frequency band [71].

Hypothesis H2 expected that the MG21 might be able to connect to frequency bands other than the ones observed in the TEMS reading. The MG21 is expected to have better antenna radiation, enabling it to reach cells that the TEMS device and the smartphone couldn't. However, Hypothesis H3 was disproven due to the fact that the smartphone also connected to the 2100 MHz and 900 MHz frequency bands, although it had very few data points associated with these frequency bands, as shown in Figure 4.1b.

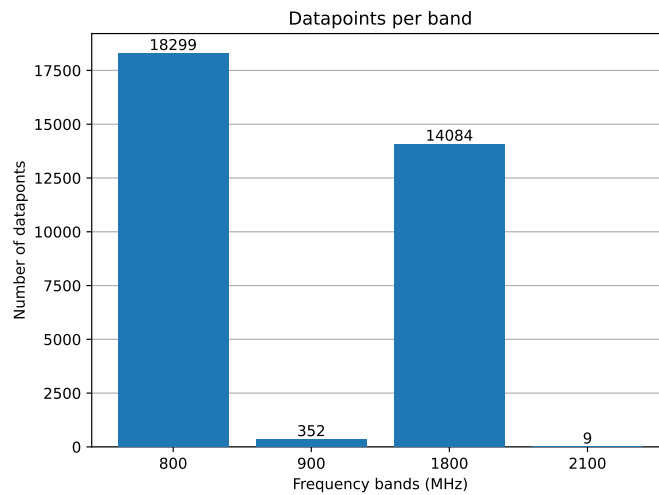
Another possible reason why the 900 MHz and 2100 MHz frequency bands did not appear in the TEMS reading could be due to the fact that the TEMS result is based on a single drive-through in one direction only. In comparison, this project conducted multiple tests in each direction, resulting in more comprehensive data for the specific test area.

Figure 4.1a shows that the capacity bands were used slightly more frequently than the coverage bands on the MG21. Meanwhile, Figure 4.1b demonstrates that the coverage bands were used more than the capacity bands on the smartphone. This could be because of the antenna radiation for the MG21 being better than

the smartphone, which can lead to the capacity bands being more available and reliable for the MG21.



(a) Datapoints per frequency band for the MG21



(b) Datapoints per frequency band for the smartphone

Figure 4.1: Amount of datapoints on the different frequency bands throughout all the tests from both test devices

4.2.2 Frequency band patterns

It was noticed that the MG21 and the smartphone formed patterns for when the different frequency bands was used throughout the test area. This section will take a look at how each device uses the frequency bands throughout the test area and compare the frequency band patterns of the two UEs.

MG21 Frequency pattern

The pattern that forms from where the MG21 uses the different frequency bands is similar in most of the tests. A typical example of what that pattern looks like is what happened during test 3 on 13.03, as shown in Figure 4.2. Figure 4.2a shows that when driving north the 1800 MHz frequency band was used more frequently than the 800 MHz frequency band on the MG21. Figure 4.2b does show that the 800 MHz frequency band was used more than the 1800 MHz frequency band when driving south. Even though it is shown in the result that the 1800 MHz frequency band is also available throughout the whole test area, did the MG21 choose to stay on the 800 frequency band while driving south for the majority of the test. driving direction

Hypothesis H6 expected that the MG21 would connect to different frequency bands depending on the driving direction. It was pointed out earlier that the MG21 is a stationary device, one can assume that MG21 might struggle to switch between cells while in motion. Figure 4.2 indicates that the MG21 was hesitant to switch between frequency bands when driving in a certain direction. Since there is no access to information regarding the cells or eNodeBs to which the MG21 was connected to at all times, it is impossible to confirm the validity of Hypothesis H6.

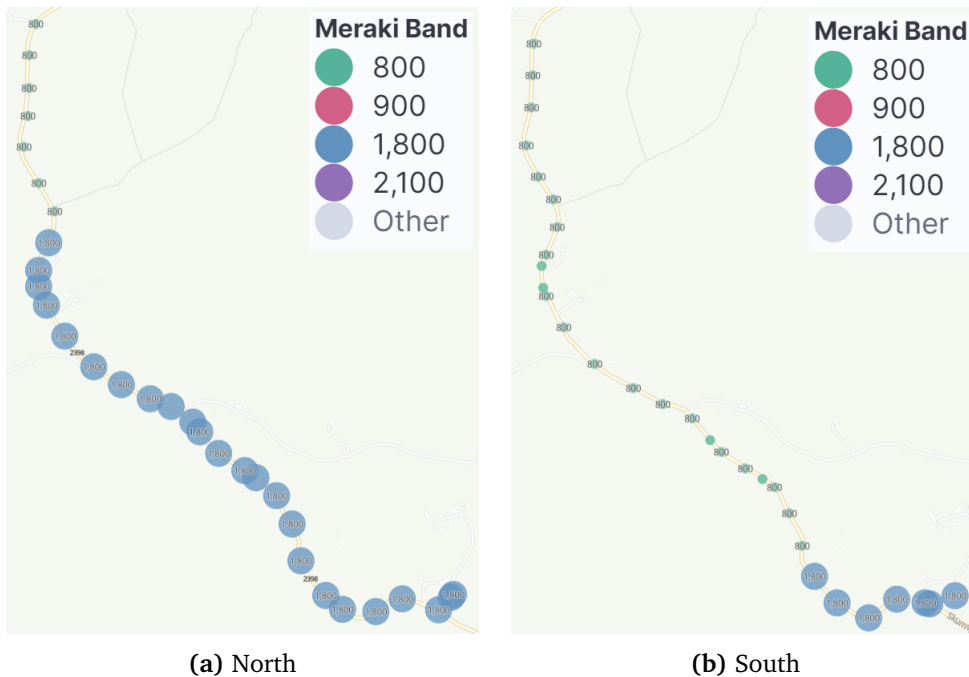


Figure 4.2: Frequency bands used by the MG21 during test 3 on 13.03 in each driving direction.

There is also a lot of inconsistency with this pattern in Figure 4.2. There were test rounds where the pattern wasn't followed at all, and the 900 MHz and 2100 MHz frequency bands were introduced. Figure 4.3 shows the highest frequency band registered within the test area for each direction. The capacity band seems to be available in the majority of the test area from both driving directions, which the test from 13.03 didn't capture.

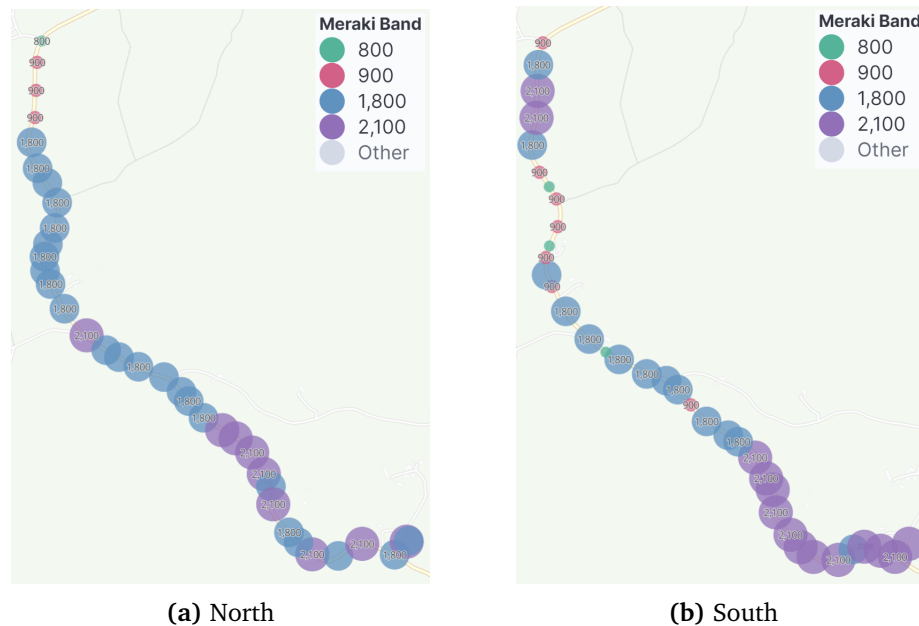


Figure 4.3: Highest frequency band used by MG21 in each section of the test area for both driving directions

It has not been pinpointed as to the specific reason for these inconsistencies regarding the frequency band used by the MG21. Telenor's TEMS readings (Section 3.4) have documented inconsistencies in signal quality within the test area. One can speculate that these inconsistencies may be due to variations in the signal parameters RSRQ and RSRP during frequency switches, which occur each time the stretch is driven. These inconsistencies could occur when different frequency bands are utilized. The discussion on RSRQ and RSRP results, along with their correlation to frequency switches, will be presented Section 4.3.

Smartphone frequency band pattern

When comparing the frequency band used by the smartphone with the frequency band used by the MG21 throughout the test area, many similarities in usage were observed. Figure 4.4 shows the results from the smartphone in the same test as the one depicted in Figure 4.2. When comparing the results in the northern region for both devices, they appear to follow almost the same pattern. Both devices

connected to the 1800 MHz frequency band initially and then switched to the 800 MHz frequency band in the southern half of the test area. While driving south, the results for both devices differed slightly, with the smartphone using the 1800 MHz frequency band more frequently than the MG21. However, both devices ended up using the 1800 MHz frequency band.

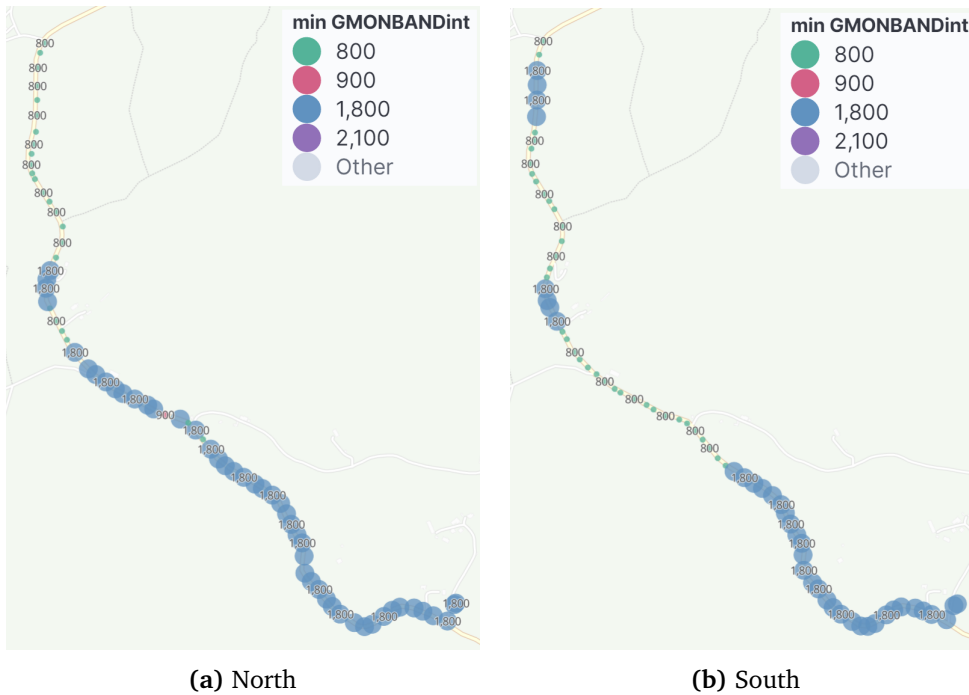


Figure 4.4: Frequency bands used by the smartphone during test 3 on 13.03 in each driving direction.

Hypothesis H5 has been debunked, as it has been discovered that the driving direction does indeed affect the frequency band used by the smartphone. This finding aligns with a possible explanation for the inconsistency observed in MG21, suggesting that the signal parameters may have interfered with the smartphone's frequency switch decisions during each drive. Additionally, more thorough signal measurement readings of the test area have been conducted compared to Telenor's TEMS readings.

Figure 4.5 shows a similar inconsistency as the MG21 when it comes to the highest frequency band registered throughout all the tests from smartphone. Figure 4.5a shows that when driving north, the capacity bands are available throughout the majority of the test area, except for the southernmost part where the 800 MHz frequency band is the only one available for the smartphone. When driving south with the smartphone, there were parts of the test area where the capacity bands were not available, but within those same sections along the stretch, the capacity bands were available for the MG21, as shown in Figure 4.3b. This might

be because the radiation pattern and gain in the MG21 antenna differ from those of the smartphone antenna, leading the MG21 to have a better signal to the cells from the capacity bands. In Hypothesis H1, it was speculated that the MG21 would connect to the 1800 MHz frequency band in more places than the TEMS device did, due to the possibility that the MG21 antenna might give it an advantage over the smartphone in terms of availability of the capacity bands. However, the smartphone also received coverage from the 1800 MHz frequency band more frequently than mentioned in the TEMS readings. This has weakened Hypothesis H1. Nonetheless, the fact that the 2100 MHz frequency band is much more available on the MG21 might still prove the point that capacity bands are more accessible due to the MG21's superior antenna radiation.

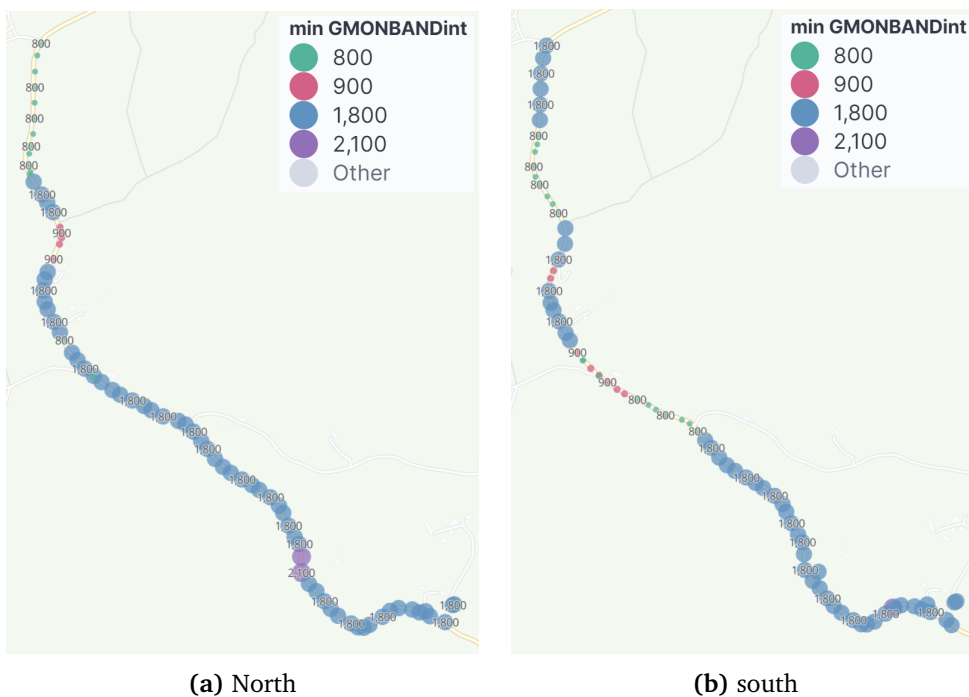


Figure 4.5: Highest frequency band used by the smartphone in each section of the test area for both driving directions

4.2.3 Occurrence of the 900 MHz frequency band

While the 900 MHz frequency band was not the focus of this report, test 3 on 17.03, in particular, stands out from the rest. Disregarding outliers, the MG21 starts a round of tests on the 1800 MHz frequency band and changes to the 800 MHz frequency band while driving north, as can be seen in Figure 4.2. Then, as the car is going back south, it would swap back to the 1800 MHz frequency band. However, in this test, while driving the third round going north, the MG21 changes to the 900 MHz frequency band instead of the 800 MHz frequency band. This has occurred for a couple of measurements during test 1 on 26.04 as well,

with the main difference being how long the MG21 stayed on the 900 MHz frequency band. Test 1 on 26.04 measured 13 data points on the 900 MHz frequency band, while test 3 on 17.03 measured 31 data points. Together, these tests cover all 44 measurements the MG21 has on the 900 MHz frequency band during all the tests, as can be seen in Figure 4.1a.

During test 1 on 26.04, the MG21 changed to the 900 MHz frequency band while driving south. It had been on the 800 MHz frequency band since driving north. In contrast, during test 3 on 17.03, the MG21 changed to the 900 MHz frequency band while driving north. It only measured a single point of data on the 800 MHz frequency band after dropping down from the 1800 MHz frequency band. The MG21 then remained on the 900 MHz frequency band for all but the last measured data point of the test, which was back on the 1800 MHz frequency band. Figures 4.6 and 4.7 show the difference between the frequency bands used by the MG21 while driving north and south for both round 1 and round 3 during test 3 on 17.03.



(a) Frequency bands used on the first round driving north (b) Frequency bands used on the third round driving north

Figure 4.6: Comparison between frequency bands used by the MG21 during round 1 and 3 from test 3 on 17.03 while driving north.

NOTE: Kibana is set to display the minimum value if there are overlapping points on the map.

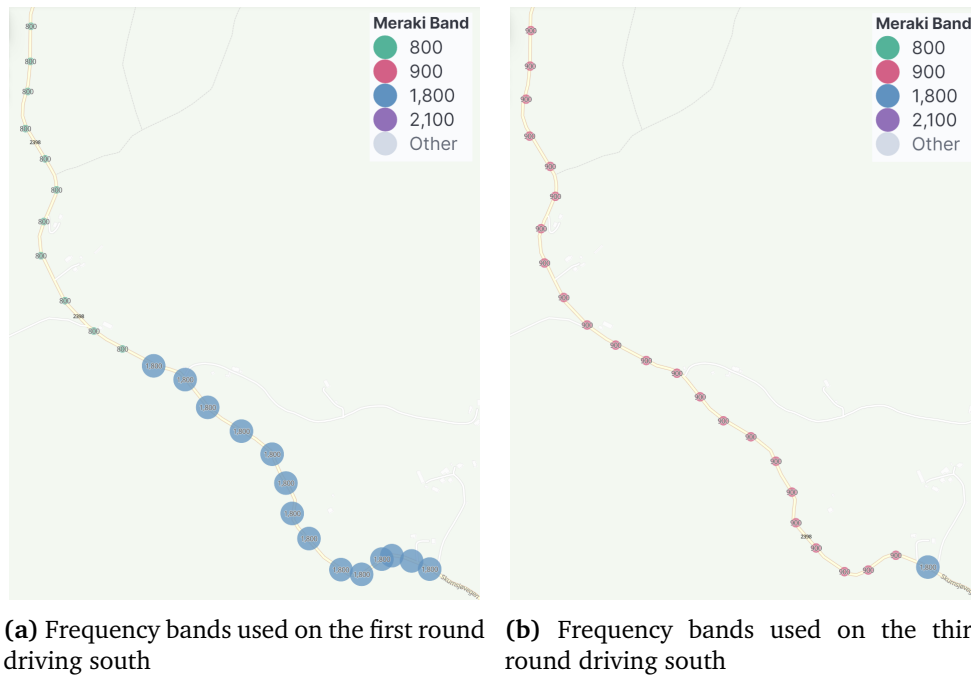


Figure 4.7: Comparison between frequency bands used by MG21 during round 1 and 3 from test 3 on 17.03 while driving south.

NOTE: Kibana is set to display the minimum value if there are overlapping points on the map.

Looking at the frequency band usage from G-MoN in figure 4.1b, one can observe that the smartphone also occasionally connects to the 900 MHz frequency band. Unlike the MG21 though there are only a couple of tests where the smartphone does not connect to the 900 MHz frequency band at all. However, while the smartphone connects to the 900 MHz frequency band in more tests, it often does not use the frequency band for more than a handful of measurements, ranging from two measurements to 32, with the outlier being test 1 on 26.04 with 269 measurements. Figure 4.1b shows that the smartphone is connected to the 900 MHz frequency band for a total of 352 measurements, making test 1 on 26.04 the source of the majority of these measurements. However, considering that the average amount of entries from the G-MoN logs are 3628, it means that except for test 1 on 26.04, the smartphone is using the 900 MHz frequency band for less than 0.1% of the measurements. There does not seem to be any distinguishable pattern for when the smartphone uses the 900 MHz frequency band as it happens both while driving north and while driving south. Furthermore, there is no clear correlation between when the smartphone and the MG21 are using the 900 MHz frequency band. They are connected to the 900 MHz frequency band at the same time for a couple of minutes during test 1 26.04, but not at all on test 3 17.03.

One possible reason for the MG21 connecting to the 900 MHz frequency band during these test 3 on 17.03 and test 1 on 26.04 is that it performs a handover to a different eNodeB than it usually does. Due to the Cisco Meraki Cloud and the Cisco Meraki RESTful API's inability to display the connected eNodeB for the MG21, it is not possible to determine if it is connected to an eNodeB it doesn't typically connect to. However, G-MoN on the smartphone tracks the currently used eNodeB, revealing that the smartphone connects to two different eNodeBs while operating on the 900 MHz frequency band. These eNodeBs also uses other frequency bands, and the smartphone connects to them at different times when not connected to the 900 MHz frequency band. Another possible explanation for the MG21 and the smartphone connecting to the 900 MHz frequency band is that the serving eNodeB in the 800 MHz frequency band reaches maximum capacity due to network congestion. Network congestion was something that was expected to affect the capacity with Hypothesis H8, but since there wasn't any equipment to monitor the traffic flow on Telenor's 4G network, this will be based solely on speculations. Consequently, the eNodeB may redirect the MG21 to the 900 MHz frequency band instead. However, this does not explain why the MG21 does not revert back to the 1800 MHz frequency band while driving south, as it does during round 1 and in most other tests.

4.3 RSRP and RSRQ

RSRP and RSRQ are both signal parameters that are important for the UE when considering which frequency band to use and whether to perform a handover or not. This section will take a look at the pattern of RSRP and RSRQ and how they behaved in certain situations throughout the tests.

4.3.1 MG21 with stronger RSRP

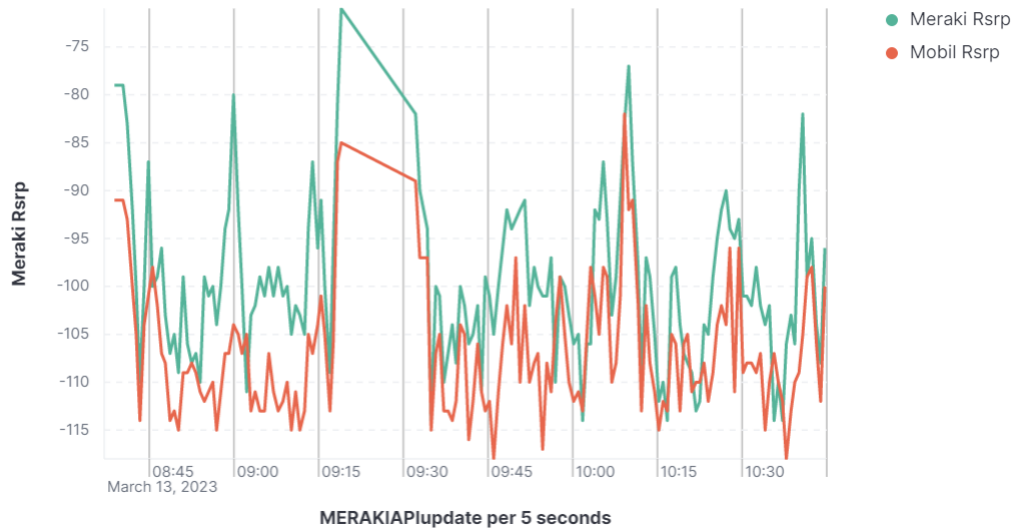
From all the results that were gathered from the MG21 and the smartphone used, the RSRP and RSRQ values from both devices followed the same trends when a change occurred with these signal parameters. This means that when the RSRP was dropping, the RSRQ dropped down relatively in the same trajectory for both test devices, as seen in Figure 4.8. Hypothesis H11 did not anticipate this occurrence, considering the relationship between RSRP and RSRQ calculations.

An interesting observation is that the MG21 had a higher RSRP value than the smartphone most of the time. An interesting observation is that the MG21 had a higher RSRP value than the smartphone most of the time. This is not surprising due to the expectation in Hypothesis H12 where the MG21 has stronger radiation with its antenna than the smartphone, which enables MG21 to have a stronger signal quality. However, Hypothesis H12 did not expect that the opposite would happen when looking at the RSRQ result, where the smartphone had a higher value than the MG21 most of the time. This was surprising because when meas-

uring RSRQ, RSRP is one of the factors considered for calculating the value of RSRQ, as well as any interference that is affecting the radio signal. The reason behind this could be that the MG21 can be more prone to interference than a smartphone. Since the MG21 is meant to be a stationary device, it may not be able to adapt to receive the radio signal while it's on the move. Therefore, interference can affect the MG21 more than a smartphone, which is made to be mobile. Another possible reason could be that since the MG21 has a better average RSRP, indicating that it has a better antenna and better signal reception. This antenna may also be better at listening to the interference in the environment, which could offset the better average RSRP, resulting in a lower average RSRQ compared to the smartphone.

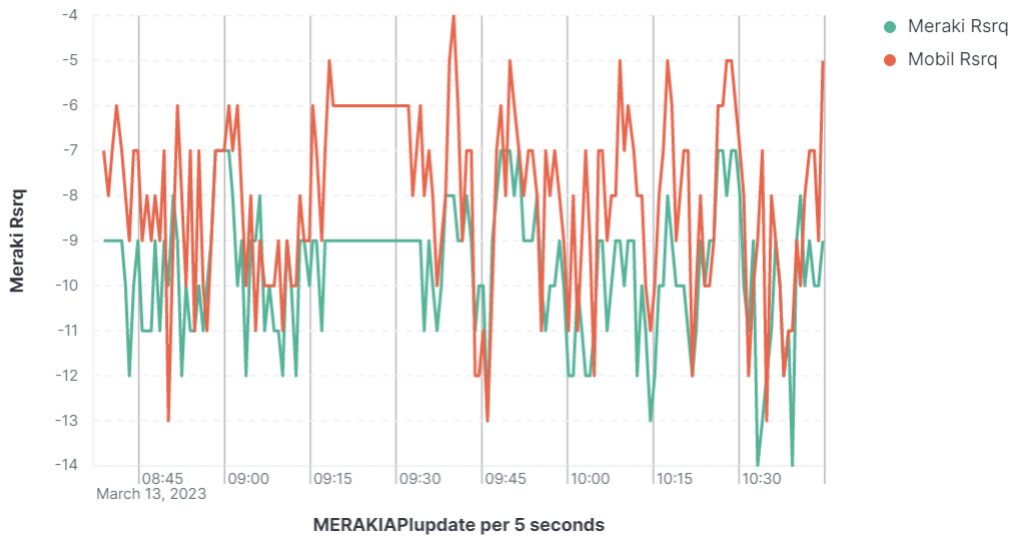
Any equipment to measure the available signal interference that is present in the environment wasn't available throughout these tests. Therefore, there is no certainty that the only factor for a lower RSRQ from the MG21 is because of it being more prone to interference.

Meraki vs Mobil Rsrp



(a) RSRP values for first test

Meraki vs Mobil Rsrq



(b) RSRQ values for first test

Figure 4.8: RSRP and RSRQ values from both the smartphone (red) and the MG21 (green) taken during test 1 13.03. The flat line in both figures between 09:15 and 09:30 is because of missing data points from a small break in the test.

Table 4.1 shows the average RSRP and RSRQ gathered from the tests, indicating that the MG21 had decent values, despite the test area being reported as having inconsistent/bad coverage. The average RSRP from the MG21 ranged from -98 dBm to -101 dBm, with an average RSRQ between -8 dB and -13 dB. According to

how Cisco Meraki gateways interpret RSRP and RSRQ values (Table 3.2), the average signal strength would correspond to 3-4 signal bars. In most cases, a 3-4/5 signal bar rating indicates acceptable signal quality for most UEs. As mentioned earlier, the RSRP and RSRQ values were generally lower for the smartphone results. However, since UEs interpret RSRP and RSRQ values differently, one cannot be certain if the smartphone values were acceptable or not.

4.3.2 Varying values within geographical points

Throughout all the tests, the RSRP and RSRQ from each geographical point had instances where the values were varying each time the road stretch was driven. This means that when driving North once, a set of RSRP is gathered that is a bit different from when driving the same route for a second and third time. This was a recurring thing happening throughout the majority of the tests that were taken, no matter which driving direction the vehicle was headed. Figure 4.9a shows the maximum RSRP registered in each geographical point based on all the tests performed driving south, while Figure 4.9b shows the minimum RSRP registered for the same criteria. The difference between the best and worst RSRP values for the same geographical points was noticeable, with a difference of as much as 27 dBm between the best and worst RSRP. The same observation was noticed on the RSRP and RSRQ gathered from the smartphone.

The cause of this could be that the different factors that can affect the radio signal may be different from time to time. For example, the interference that is present at a certain location can be different each time. Another factor is that the network traffic (Section 2.1.5) on the 4G network can be different at different times of the day.

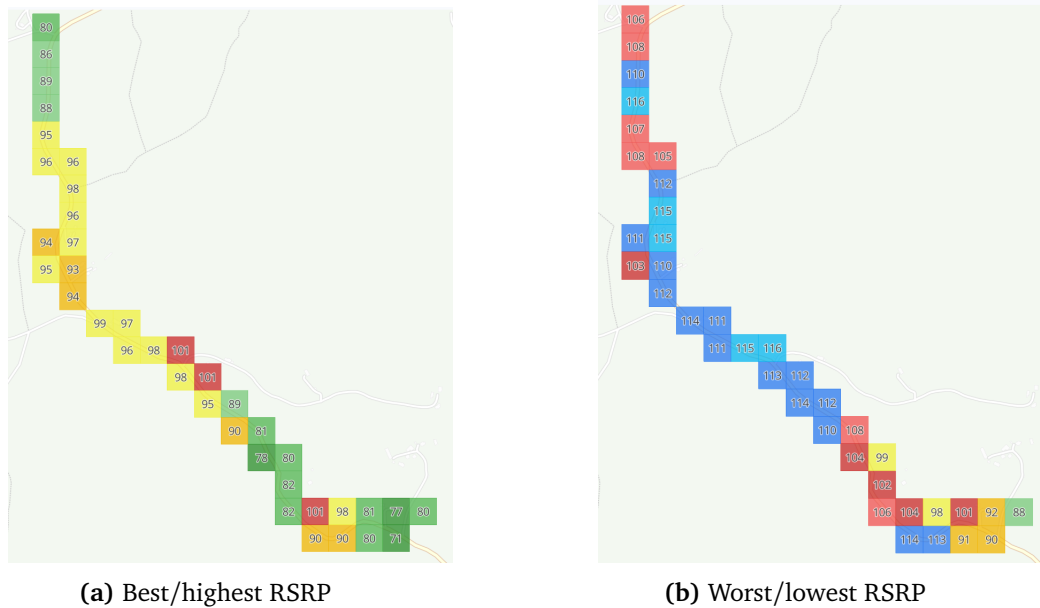


Figure 4.9: Comparison between the best and worst RSRP values in each geographical position gathered from the MG21 throughout all tests driving south

4.3.3 Weather

Hypothesis H14 expected that weather would have an effect on the 4G signal quality. The results in Table 4.1 show that the average RSRP and RSRQ values are fairly similar. On the MG21, the average RSRP ranges from -98.27 dBm to -101.35 dBm, and the average RSRQ ranges from -9.6 to -13.29 dB. However, when examining specific tests in relation to weather conditions, the data reveals that days with the heaviest snowfall did not necessarily exhibit the worst average values. For example, test 2 on 13.03, which experienced the most snow, had an average RSRP of -98.76 dBm and an average RSRQ of -9.62 dB. In comparison, test 2 on 17.03, which had no snow, resulted in worse values. Similar trends are observed when considering the minimum and maximum values. While the performance on days with heavy snowfall is not the best, it is also not the worst. The results from G-MoN on the smartphone exhibit a similar pattern to those of the MG21. Overall, it appears that the amount of snowfall did not have a noticeable effect on the RSRP and RSRQ values. Although Hypothesis H14 has been weakened in our tests, it may still hold true under different weather conditions and with a larger number of tests to gather more data.

4.3.4 RSRQ inconsistency

One interesting observation of RSRQ is that for the tests on 26.04, the average RSRQ on the MG21 was significantly worse than on the other days tested as can be seen in Table 4.1b. It was not isolated to just one of the tests either, as both

of the tests done on 26.04 shows similar average RSRQ. The average is -13.29 dB for the first test and -12.96 dB for the second test, while the worst average observed before this was during test 3 on 17.03 which had an average -10.35 dB. Considering the test results show RSRQ being in the -7 dB to -17 dB range on the MG21, having a decrease of 2.94 dB and 2.61 dB from the previous worst result is significant.

A similar reduction in average RSRQ can also be observed in the results from G-MoN on the smartphone, with an average of -10.54 dB on the first test and -11.38 dB on the second test. While not as large of a decrease as the MG21, the smartphone experienced a decrease of 1.45 dB and 2.29 dB respectively compared to the previous worst result of -9.09 dB during test 3 on 17.03.

The tests performed on 26.04 were done using the same hardware setup as previous tests. The main difference in our software was measuring download speed with iPerf3 rather than upload speed. This should not make any difference as UL and DL are affected by RSRQ and RSRP, not the other way around. The only other noticeable difference between these two tests and the other seven tests was the environment conditions. While it was neither snowing nor raining, the trees were covered in heavy snow from the day before. The tests on 13.03 and 17.03 also had snow on the trees and snowfall, however they had light more powder-like snow. The heavier more dense snow on the trees could be a possible explanation for the increase RSRQ as denser snow could increase the interference (Section: 2.1.4) on the radio signal, due to the increased path obstruction. This can also explain why only the average RSRQ is lower and not the RSRP, as RSRP is not affected by interference.

4.3.5 Difference between 800 MHz- and 1800 MHz frequency bands

The 800 MHz and 1800 MHz frequency bands are the main focus of this thesis. There are differences in how the two frequency bands receive RSRP and RSRQ. The following section will first take a look at the test results from 01.03 to 17.03, while the test taken on 26.04 will have a separate section due to some irregularities in the results compared to the rest of the days.

RSRQ

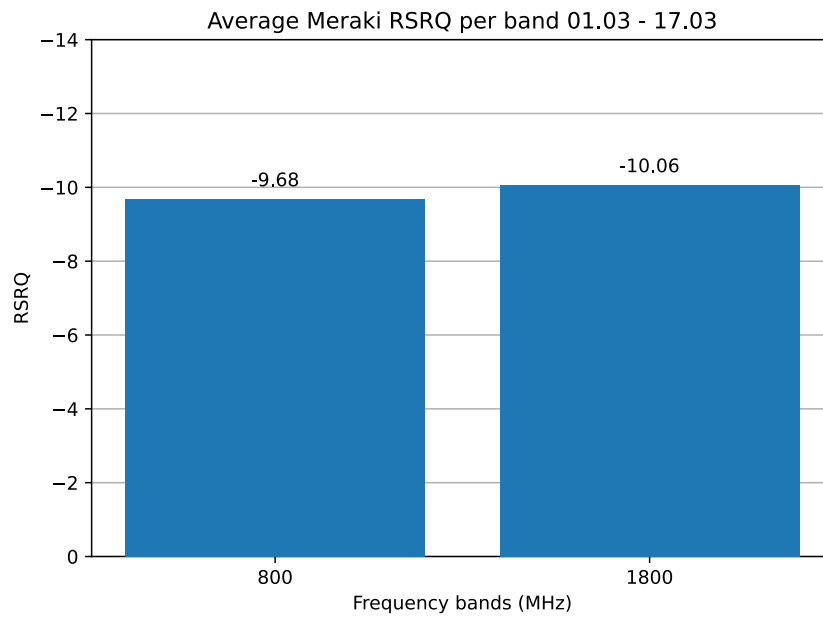
When comparing the result of RSRQ from the 1800 MHz and 800 MHz frequency bands throughout the test done from 01.03 to 17.03, the results showed a similar pattern for both frequency bands. Table 4.2 shows the maximum and minimum RSRQ values from each frequency band. There seems to be little difference between the 1800 MHz and 800 MHz frequency bands. The minimum RSRQ on the 1800 MHz and 800 MHz frequency bands is almost identical on the MG21. The same can be said about the maximum RSRQ on both the MG21 and the smartphone. There was a slight difference in the minimum RSRQ between the 1800

MHz and 800 MHz frequency bands on the smartphone, with slightly worse RSRQ on the 800 MHz frequency band.

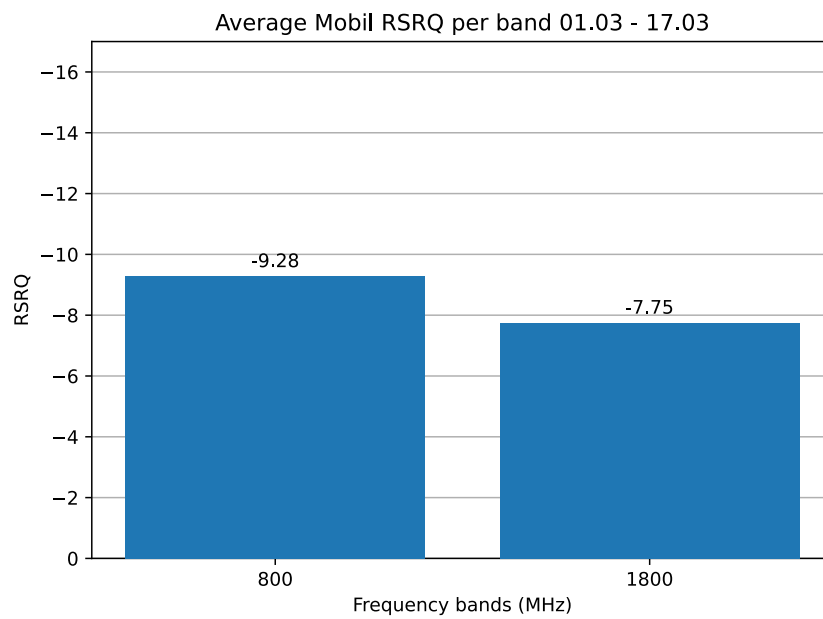
Data	01.03	13.03	17.03
1800 MHz Frequency band			
MG21 max	-8dB	-7dB	-7dB
MG21 Min	-13dB	-14dB	-15dB
G-MoN max	-4dB	-4dB	-5dB
G-MoN min	-14dB	-16dB	-14dB
800 MHz Frequency band			
MG21 max	-7dB	-7dB	-7dB
MG21 Min	-14dB	-14dB	-15dB
G-MoN max	-5dB	-4dB	-5dB
G-MoN min	-16dB	-18dB	-17dB

Table 4.2: Comparison of RSRQ between the 800 MHz and 1800 MHz frequency bands for the test done between 01.03 and 17.03. The data is from both the smartphone and the MG21

The maximum and minimum RSRQ do not showcase how RSRQ behaved in general throughout the tests, while the average RSRQ of each frequency band could demonstrate the overall difference between the frequency bands. Figure 4.10a illustrates that the 1800 MHz frequency band has a slightly worse value than the 800 MHz frequency band for the MG21. This complies with Hypothesis H13, which states that the signal parameters are better on the coverage band due to its improved coverage and penetration.



(a) MG21 RSRQ



(b) Smartphone RSRQ

Figure 4.10: The average RSRQ of the MG21 and the smartphone during the tests between 01.03 and 17.03

However, when analyzing the results from the smartphone, they don't follow the same pattern as the MG21's RSRQ values. Figure 4.10b does show that the

average RSRQ on the 1800 MHz frequency band had a better RSRQ than the 800 MHz frequency band for the tests conducted between 01.03 and 17.03. This was contrary to the expectation stated in Hypothesis H13 and also differed from the MG21's RSRQ results. This will be discussed later in Section 4.3.5.

RSRP

The RSRP result from the 1800 MHz and 800 MHz frequency bands, obtained from tests performed from 01.03 to 17.03, was slightly different from the RSRQ. While there was minimal to no difference between the maximum and minimum RSRQ values across the two frequency bands, a more noticeable difference was observed with RSRP. Table 4.3 shows that the minimum RSRP from MG21 was considerably better on the 800 MHz frequency band than on the 1800 MHz frequency band. One possible reason for this is that it is easier to spot differences in RSRP values compared to RSRQ, as RSRP has a larger range of values (-44 dBm to -140 dBm), while RSRQ ranges between -3 dB and -19.5 dB.

It is worth noting that the minimum RSRP recorded on the MG21 on 13.03 was -116 dBm. At that point, the upload speed was observed to be 0 Mbps, suggesting a possible handover event captured in that data point (more details about handover results can be found in Section 4.4). Therefore, -106 dBm can be considered the lowest RSRP measurement that is not indicative of a possible handover.

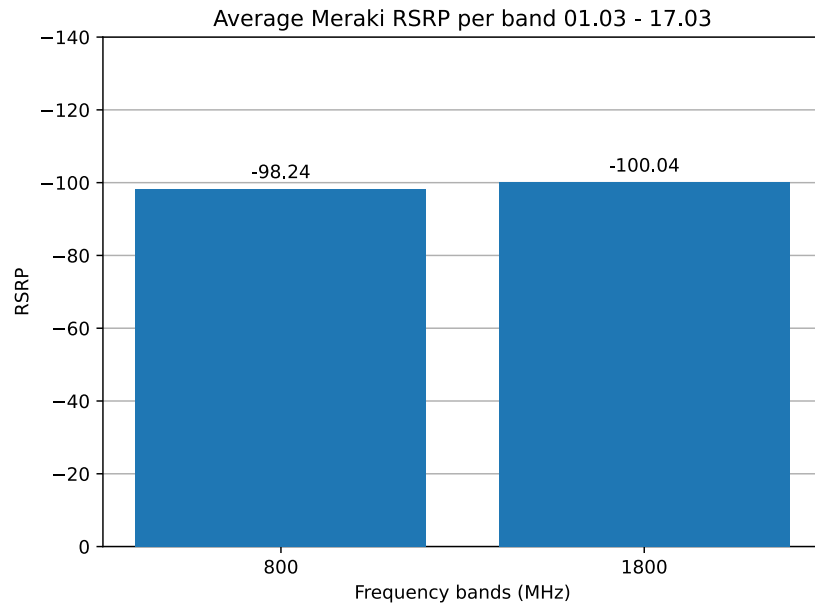
Data	01.03	13.03	17.03
1800 MHz Frequency band			
MG21 Max	-73 dBm	-77 dBm	-75 dBm
MG21 Min	-114 dBm	-115 dBm	-115 dBm
G-MoN Max	-82 dBm	-80 dBm	-83 dBm
G-MoN Min	-120 dBm	-120 dBm	-122 dBm
800 MHz Frequency band			
MG21 Max	-80 dBm	-78 dBm	-80 dBm
MG21 Min	-105 dBm	-106 dBm (-116)*	-107 dBm
G-MoN Max	-92 dBm	-90 dBm	-91 dBm
G-MoN Min	-118 dBm	-122 dBm	-123 dBm

Table 4.3: Comparison of RSRP between the 800 MHz and 1800 MHz frequency bands from the tests on 01.03 to 17.03. The data is from both the smartphone and the MG21

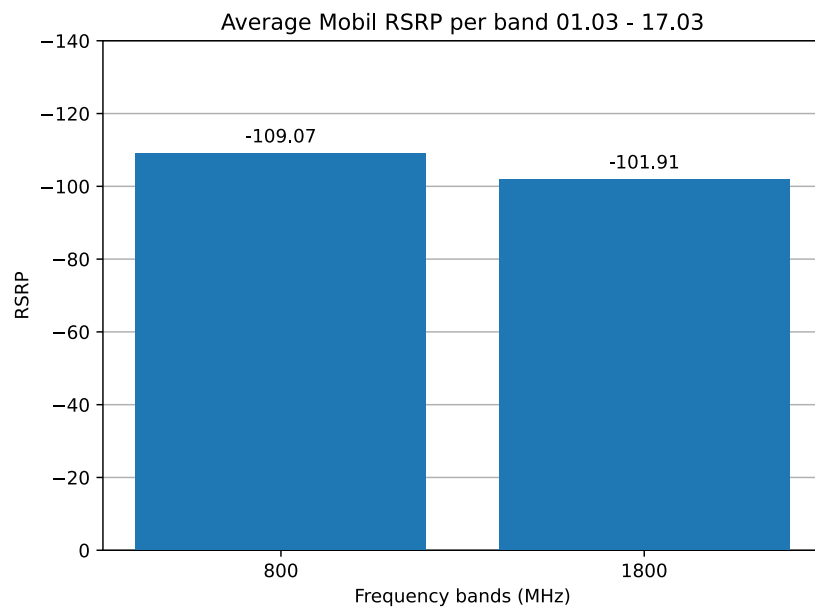
(* -116 dBm was the lowest RSRP, but that captured during a possible handover. second highest was -106 dBm and there were several data showing this value)

Figure 4.11 shows the average RSRP from the MG21 and the smartphone based on the tests conducted between 01.03 and 17.03. The average RSRP results exhibit the same pattern as the average results from RSRQ in Figure 4.10. The

smartphone demonstrates better RSRP on the 1800 MHz frequency band compared to the 800 MHz frequency band, whereas the MG21 displays better RSRP on the 800 MHz frequency band than the 1800 MHz frequency band.



(a) MG21 RSRP



(b) Smartphone RSRP

Figure 4.11: The average RSRP of the MG21 and the smartphone during the tests between 01.03 and 17.03.

Hypothesis H13 expects that the signal parameter would be better on the coverage bands than the capacity bands. This expectation is fully supported when looking solely at the MG21's overall results. However, when examining the smartphone's overall results, the opposite of what was expected from Hypothesis H13. The signal parameters were better on the capacity bands instead of the coverage bands. This will be discussed later in Section 4.3.5.

Possible reason for difference between Smartphone and MG21 result

There could be many factors for why the 1800 MHz frequency band does have better RSRP and RSRQ on the smartphone than the MG21. One would think to believe that the 800 MHz frequency band would always have better RSRP and RSRQ due to it being a coverage band and therefore providing a stronger signal for UEs. It's also important to consider when the UEs decide to switch frequency bands to the coverage bands or the capacity bands. If the UE decides to switch to the coverage band when the RSRP and RSRQ are at a certain threshold on the capacity band, then the RSRP and RSRQ won't be as low on the capacity band as it would be on the coverage band. For Telenor's network, the lowest available 4G frequency band is the 700 MHz frequency band. If the RSRP and RSRQ gets low when connected to the 700 MHz frequency band, then the UE will be stuck on that frequency band and can't switch to a lower frequency band. In the test area, there was no coverage from the 700 MHz frequency band; instead, there was coverage from the 800 MHz frequency band.

It is believed that the reason why the RSRP and RSRQ on the 1800 MHz frequency band are better on the smartphone is because the smartphone is a better device when it comes to deciding when to switch to a coverage band. The smartphone prioritizes having better signal quality over choosing a capacity band with lower RSRP and RSRQ. Figure 4.1 also confirms that the 1800 MHz frequency band was used more frequently than the 800 MHz frequency band on MG21, while the opposite occurred with the smartphone.

The MG21 is believed to have the functionality to prioritize network capacity over signal quality. It is not designed to go through several cells and eNodeBs in a short time period; instead, it aims to provide the best possible network performance for users. When the smartphone is connected to the 800 MHz frequency band and the RSRP and RSRQ become low, there is no possibility to switch to a lower frequency band since the 800 MHz is the lowest 4G frequency band allocated for Telenor's cellular network. Therefore, RSRP and RSRQ will naturally be lowest on the 800 MHz frequency band.

Interesting result from test 26.04

On 26.04, tests were performed that showed different results than the previous tests. Looking at Table 4.4, there isn't a lot of difference in the RSRP results com-

pared to Table 4.3, which shows RSRP result from the test conducted from 01.03 to 17.03.

On the other hand, RSRQ had differentiated results from 01.03 to 17.03. Table 4.5 shows that the RSRQ was generally lower on both frequency bands, especially on the MG21 where the difference was more obvious when compared to Table 4.2. The interesting part was that the lowest RSRQ on the 1800 MHz frequency band on the smartphone was lower than the lowest RSRQ on the 800 MHz frequency band, which did not happen in the previous tests.

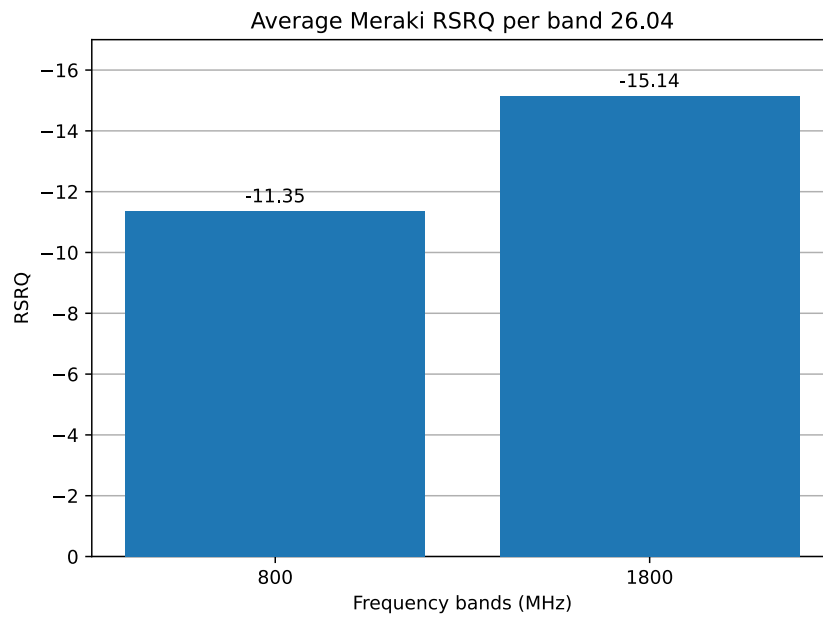
Data	26.04
1800 MHz Frequency band	
MG21 Max	-72 dBm
MG21 Min	-113 dBm
G-MoN Max	-83 dBm
G-MoN Min	-119 dBm
800 MHz Frequency band	
MG21 Max	-80 dBm
MG21 Min	-114 dBm
G-MoN Max	-96 dBm
G-MoN Min	-120 dBm

Table 4.4: Comparison of RSRP between the 800 MHz and 1800 MHz frequency bands between the test done 26.04. Test data is from both the smartphone and the MG21

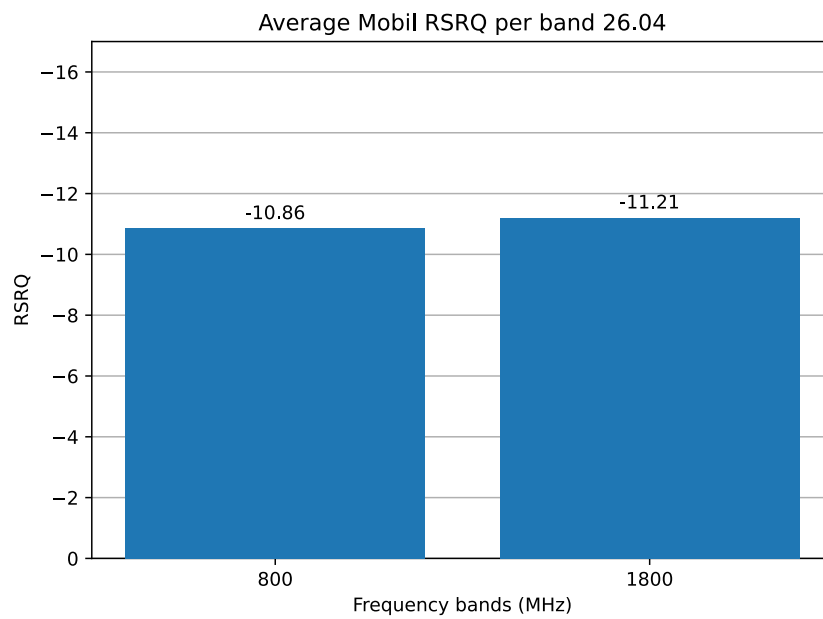
Data	26.04
1800 MHz Frequency band	
MG21 Max	-11dB
MG21 Min	-17dB
G-MoN Max	-5dB
G-MoN Min	-19dB
800 MHz Frequency band	
MG21 Max	-9dB
MG21 Min	-17dB
G-MoN Max	-6dB
G-MoN Min	-18dB

Table 4.5: Comparison of RSRQ between the 800 MHz and 1800 MHz frequency bands between the test done 26.04. Test data is from both the smartphone and the MG21

The same can be seen when looking at the average RSRQ on the smartphone in Figure 4.12. The 1800 MHz frequency band had a lower RSRQ value than the 800 MHz frequency band on the smartphone, which was not the case with the previous tests. However, on the MG21, the RSRQ followed the same pattern as the previous tests, where the 800 MHz frequency band had a better RSRQ than the 1800 MHz frequency band. The tests done on 26.04 will then comply with Hypothesis H13 for both test devices.



(a) MG21 RSRQ



(b) Smartphone RSRQ

Figure 4.12: The average RSRQ of the MG21 and the smartphone during the test conducted on 26.04

As mentioned in Section 4.3.4, the test area was surrounded by trees with heavy snow, which might have affected the RSRQ for both the smartphone and

the MG21. RSRQ considers the interference present in the environment, which is why it is lower compared to the RSRP, as the latter does not take interference into account. However, it cannot be confirmed with 100% certainty that the snow caused this interference to the radio signal, especially since interference cannot be measured with the given equipment and data.

There isn't a very good explanation as to why the average RSRQ is worse on the 1800 MHz frequency band on the smartphones compared to the 800 MHz frequency band, as illustrated in Figure 4.12b. One can assume that due to the possible high amount of interference present in the environment, smartphones may not have performed optimal frequency band switches when needed. This could have resulted in the smartphones having high values of RSRQ before they could switch to a lower frequency band. Referring to the theory in Section 2.3.2, RSRP is the main factor when performing cell selection, and RSRQ acts as a decisive factor if RSRP doesn't have sufficient values. Since RSRP did not change drastically compared to the RSRQ from previous tests, this will contradict with the assumption made early in this paragraph.

The expectation is that there are other underlying factors that affect RSRQ from the smartphone but cannot be analyzed due to the lack of equipment and data. These factors include how different types and amounts of interference impact a smartphone compared to a cellular gateway, as well as how different UEs perform cell selection when receiving high values of RSRQ. These are some of the factors that may explain this scenario.

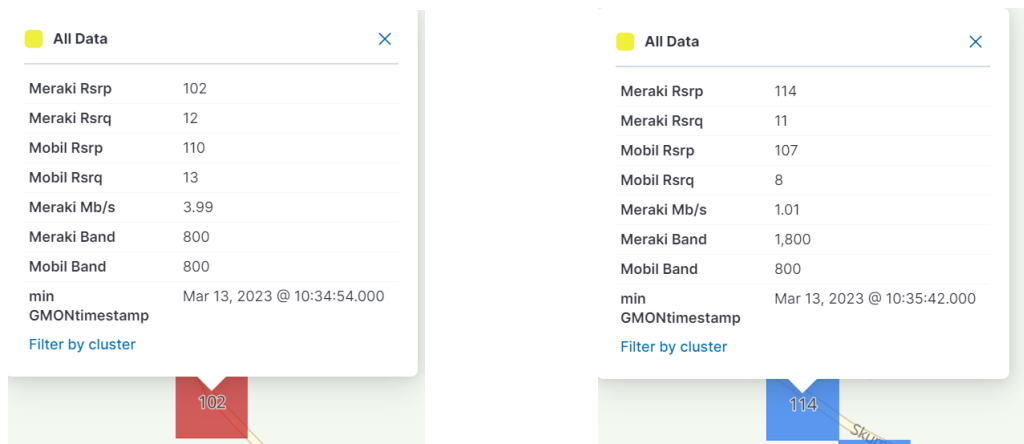
4.4 Handover and switches between frequency bands

After analyzing the MG21 data from all the tests, patterns were found in the switches between frequency bands. One observation was that when RSRP rapidly increased in value, the MG21 would switch to a frequency band with a higher frequency. A similar observation was noticed when the frequency band switch was performed the other way around. When the RSRP decreased by a significant amount, the MG21 would switch to a lower frequency band to obtain a better RSRP. This partially complies with Hypothesis H9. The MG21 rapidly increased its RSRP rather than gradually, as expected in Hypothesis H9.

After the frequency switch was performed, the RSRP would again be significantly reduced. Figure 4.13 showcases an instance where RSRP is lowered when MG21 switches to a higher frequency band while driving south.

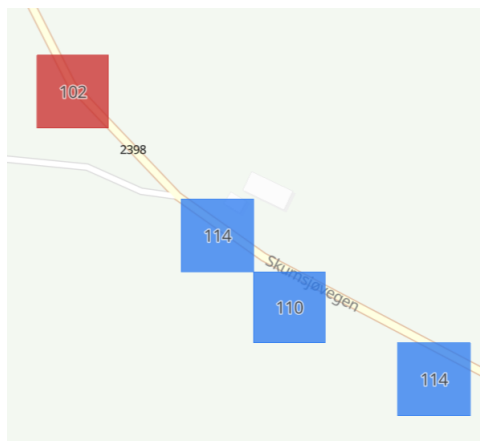
The smartphone also performed frequency band switches in a similar manner, where the RSRP would gradually drop before switching to a lower frequency band, and vice versa when the RSRP gradually increased. As opposed to the MG21, Hypothesis H9 fully aligned with the smartphone’s RSRP gradually changing before a frequency band switch.

After switching to a lower frequency band, the RSRP increased like the MG21, but the RSRP on the smartphone behaved slightly differently when switching to a higher frequency band compared to the MG21. The results showed instances where RSRP either decreased or increased after switching to a higher frequency band.



(a) Before Frequency band switch: RSRP = -102 dBm, frequency band = 800 MHz

(b) After Frequency band switch: RSRP = -114 dBm, frequency band = 1800 MHz



(c) View of where the frequency switch occurs

Figure 4.13: The MG21’s RSRP before and after the switch to a capacity/higher frequency band (driving south)

The expectation was that the RSRP would be the main factor for frequency band switches, which corresponds to the theory presented in Section 2.3. However, it was not expected for the RSRP to drop after the UEs moved to a higher frequency band. Hypothesis H10 anticipated that, following a frequency switch, the RSRP would either increase or remain the same. This is because the UEs would switch cells when the RSRP is higher than that of the current serving cell. However, the reason for this behavior across the tests remains unclear.

RSRQ didn't have a significant effect on the decisions regarding frequency band switches. The MG21 and the smartphone primarily focused on RSRP. In the smartphone results, there were a few instances where the RSRP remained the same value before and after the frequency band switch, while the RSRQ changed. This suggests that RSRQ was used in those cases to initiate the frequency band switch.

Any specific RSRP threshold for when the frequency band switch is performed was not observed from the MG21. In Figure 4.13, it was coincidental that the frequency band switched to the 1800 MHz frequency band when RSRP was -102 dBm. There were instances throughout the tests where RSRP was lower or higher than -102 dBm. The switch to lower frequency bands didn't have any specific thresholds either, but it was noticed that whenever a switch to a lower frequency band was performed, the RSRP was anywhere between -100 and -116 dBm, which is equivalent to 1 or 2 signal bars for the MG21 (MG21 signals in Table 3.2).

An observation of the RSRP from the smartphone was that the RSRP and RSRQ would gradually drop down or gradually go up, as opposed to making big jumps up and down. This was not the case with the MG21, where the RSRP and RSRQ values changed significantly within short periods of time. The MG21 may have also changed its values gradually and performed handovers before the measured data points. However, due to the lack of updates gathered from the Meraki cloud, it cannot be confirmed or denied if the values changed as gradually as the smartphone did. Therefore, it was easier to observe when the switch to another frequency band or a handover to an eNodeB was performed from the smartphone's perspective.

4.4.1 New eNodeB's discovered

Through the G-MoN log, it is possible to see to which eNodeBs the smartphone is connected at any given time. In Section 3.4.2, it is shown that TEMS readings only discovered three eNodeBs that provide 4G coverage to the test area. However, through the results from all the tests done during this project, two more eNodeBs were discovered. The smartphone was able to connect to the eNodeBs in the locations called "Helset" (D) and "Skumsjøen" (E). The placement of these two eNodeBs, relative to the test area and other eNodeBs that provide 4G cover-

age, is illustrated in Figure 4.14. The newly discovered eNodeBs provide coverage in the 1800 MHz frequency band, and the eNodeB on Helset also provides coverage in the 800 MHz frequency band for the test area. It was discovered from the G-MoN results that the 1800 MHz frequency band coverage was provided by various eNodeBs, while TEMS reading from Telenor revealed that the sole source of the 1800 MHz frequency band coverage was the eNodeB located on Skumsjovegen. The TEMS reading also detected coverage for both the 800 MHz and 1800 MHz frequencies in the test area. All the eNodeBs shown in Figure 4.14 provided 1800 MHz coverage for the smartphone. The 900 MHz coverage originated only from the eNodeBs located at Lauvkampen and Vardalsaaasen, while the 2100 MHz frequency band coverage came from the eNodeB on Skumsjovegen.

The reason why these eNodeBs and frequency bands were discovered might be the same as mentioned in Section 4.2, regarding how the 900 MHz and 2100 MHz frequency bands were found. More thorough data was gathered from these tests compared to the TEMS readings, which enabled the smartphone to discover new cells.

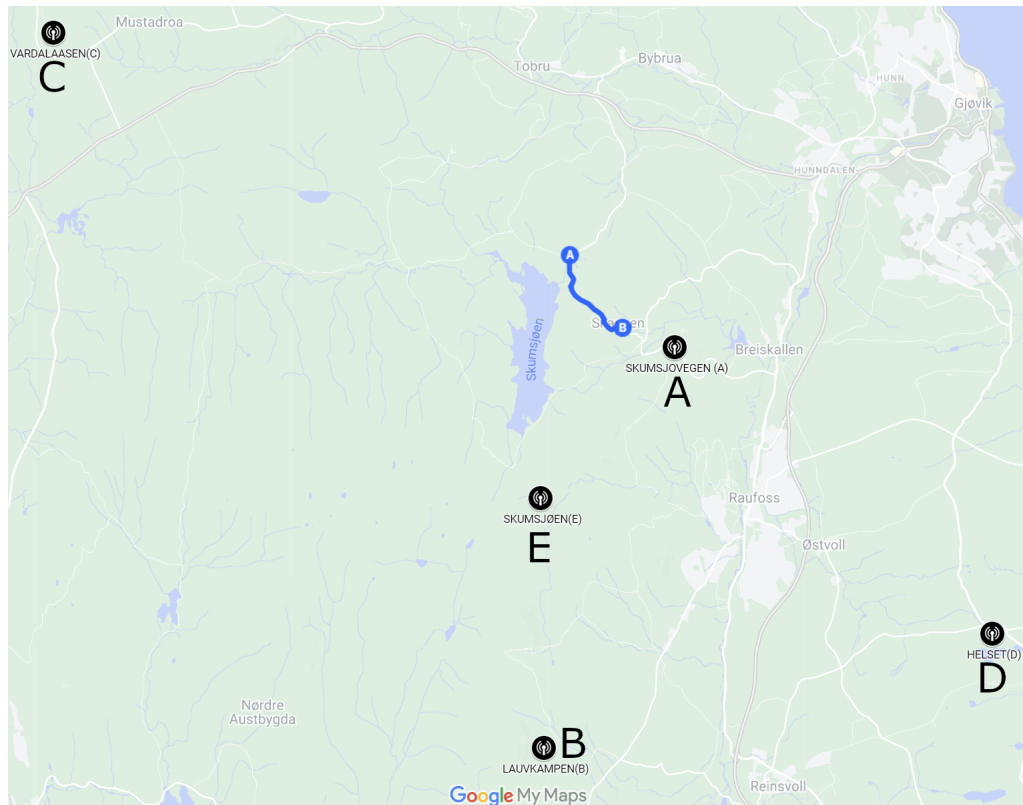
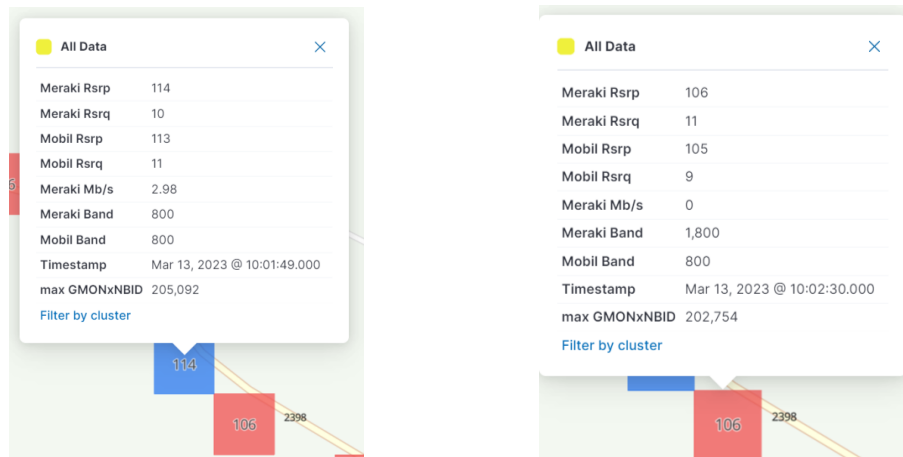


Figure 4.14: eNodeB's that was used by the smartphone during all the tests. Skumsjoevegen (A), Lauvkampen (B), Vardalsaaasen (C), Helset (D), Skumsjøen (E). Blue marking from A to B is the Test area. Taken from Google Maps

4.4.2 Bandwidth during handover

During handovers the bandwidth for UL and DL drops lower, while it is hard to pinpoint exactly when a handover occurs in the combined logs because of infrequent Meraki API updates. It is assumed that when the iPerf3 upload speed drops to 0 Mbps, it indicates that the specific data point was captured during a handover, due to the handover interruption time (Section 2.4.3). In most tests, this coincided with a Meraki API update one to three times, except for the tests conducted on 01.03 and 26.04. The test on 01.03 had no 0 Mbps measurements lining up with the Meraki API updates, and the tests on 26.04 were measuring download speed instead of upload speed. However when looking at the log files from iPerf3 rather than the combined logs it is occurring several times more for each test, for example 0 Mbps was logged 45 times during the test on 01.03.

As expected, the handovers did not always occur at the same time for both the MG21 and the smartphone. This is probably due to the different radiation patterns and gains of their respective antennas. There were very few instances where the handover may have occurred simultaneously for both devices. In those instances, it could have been purely coincidental that the handovers happened at the same time. Figure 4.15 shows an instance where the G-MoN data indicates a handover from the eNodeBs at Vardalsaasen to Skumsjovegen, while the upload speed from the MG21 was at 0 Mbps. However, one cannot be 100% sure that a handover occurred for the MG21 at that precise moment. As mentioned in the preceding paragraph, the handover interruption time can be a valuable indicator that a handover occurred for both devices.



(a) Before handover: eNodeB on Vardalsaasen and upload speed = 2.98 Mbps

(b) After handover: eNodeB on Skumsjovegen and upload speed = 0 Mbps

Figure 4.15: Possible handover occurring at the same time between the MG21 and the smartphone

Unlike upload speed, the download speed never drops all the way to 0 Mbps. It does however drop fairly low at times, particularly while on the 800 MHz frequency band, but the download speed on the 1800 MHz frequency band also drops low, often before the MG21 changes to the 800 MHz frequency band. A likely reason for the download not dropping all the way to 0 Mbps is that the MG21 supports CA on download only (Section 3.3.1). Making it so that the MG21 often has a connection to at least one eNodeB, even during a handover from one eNodeB to another, is important. However, the upload speed, for which it does not support CA, will temporarily drop to 0 Mbps due to the handover interruption time (Section 2.4.3). These drops in upload and download speed are directly related to the handover, aligning with the Hypothesis H4 which stated that the bandwidth will be affected during handovers. Consequently, the user experience will be negatively affected due to the reduction in upload and download speeds.

4.5 Upload speed

The typical pattern for the upload speed is shown in Figure 4.16, with driving direction north depicted in Figure 4.16a on the left, and south depicted in Figure 4.16b on the right. The circles in the figures represent the upload speed, where smaller and redder circles indicate poorer upload speed, while bigger and greener circles indicate better upload speed. There is no significant difference observed between the north and south driving directions. The upload speed was measured during the first seven tests conducted between 01.03 and 17.03.

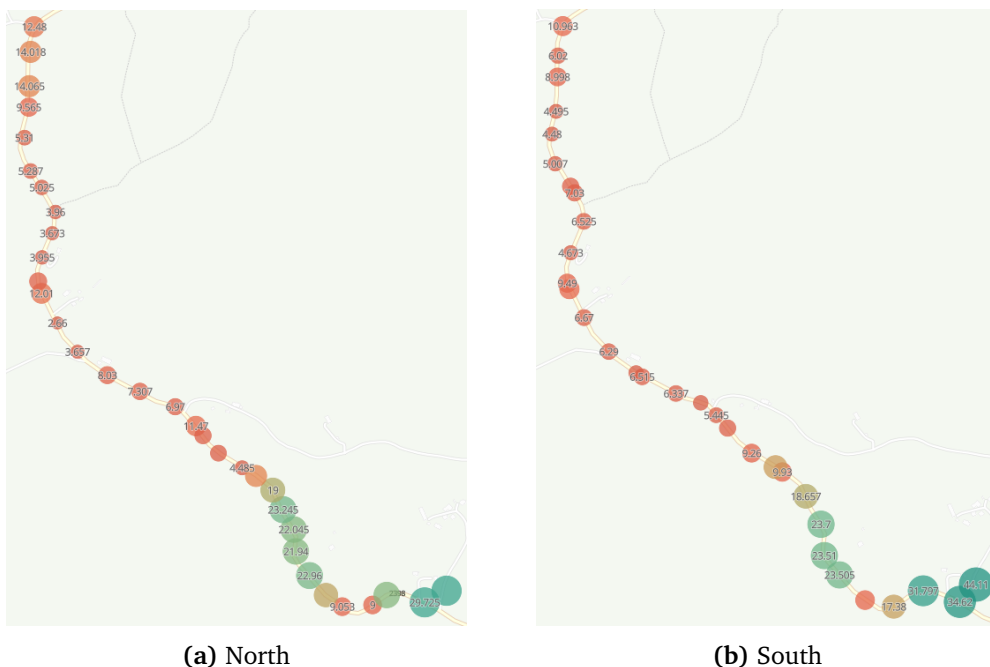


Figure 4.16: The MG21's upload speed (Mbps) measured during the test on 01.03

4.5.1 Frequency band and Bandwidth

As mentioned in Section 2.1.6, frequency bands with a frequency less than 1 GHz are considered coverage bands that provide good coverage but not as great capacity, while frequency bands with a frequency over 1 GHz are considered capacity bands that have good capacity but not as good coverage. This was confirmed during the tests, as shown in Figure 4.17. This shows that the frequency bands with higher frequency have substantially higher upload speeds than the lower frequency bands. In the data, the 900 MHz frequency band had a lower speed than the 800 MHz frequency band, but this is most likely because of a lack of data for the 900 MHz and the 2100 MHz frequency bands, as seen in Figure 4.1a. A possible reason for the 900 MHz frequency band having a lower average speed than the 800 MHz frequency band, could be because the MG21 was using the 900 MHz frequency band for an entire round. Meaning that the signal conditions were likely not optimal for the 900 MHz frequency band.

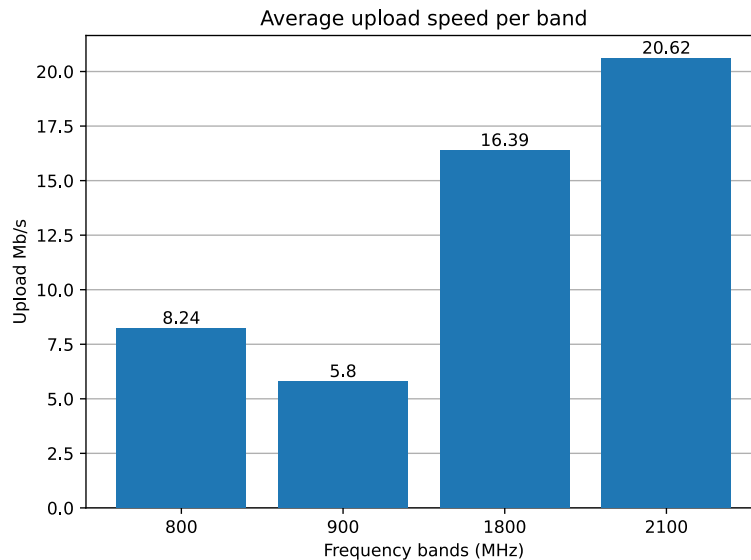


Figure 4.17: Average upload speed for the MG21 for each frequency band

Figure 4.18 shows the upload speed, RSRP, and frequency band for the first two tests on 17.03. It exhibits the same trend as the average shown in Figure 4.17: lower upload speeds on coverage bands and higher upload speeds on capacity bands. Furthermore, Figure 4.18 demonstrates the impact of RSRP on upload speed. Generally, a higher RSRP results in better upload speeds compared to a lower RSRP on the same frequency band. For instance, when the MG21 utilizes the 800 MHz frequency band and maintains a good RSRP, the upload speed can be higher than when the MG21 uses the 1800 MHz frequency band with a poor RSRP, disproving Hypothesis H7. An example of this is depicted in Figure 4.19.

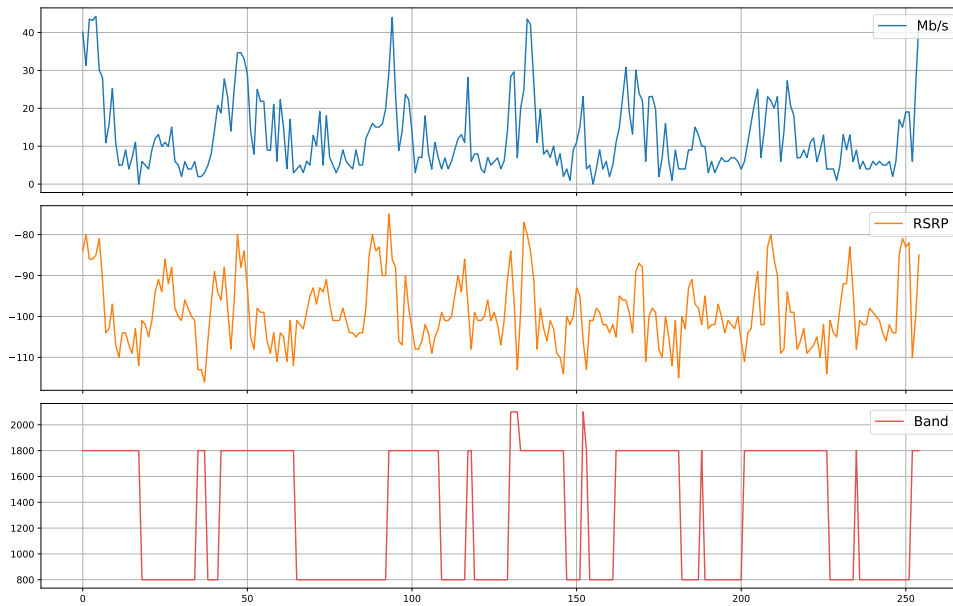
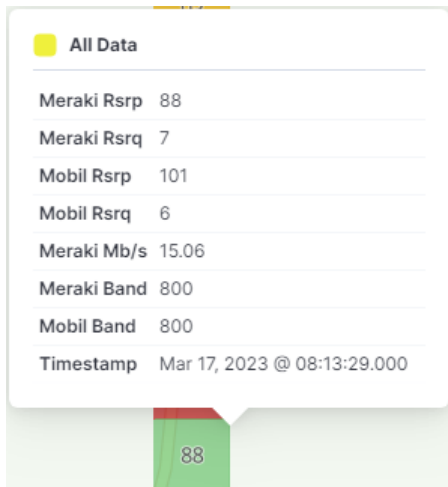
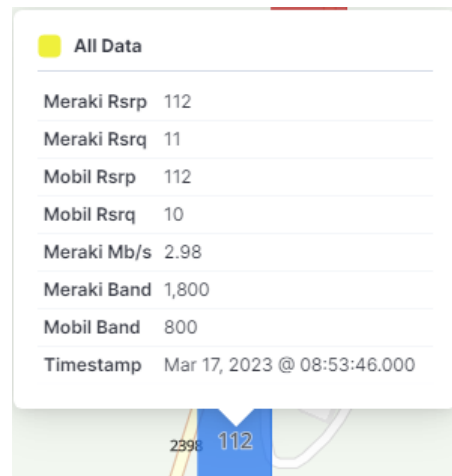


Figure 4.18: The MG21’s upload speed, RSRP and frequency band for the first two tests on 17.03.



(a) Data point on the 800 MHz frequency band with good RSRP and UL



(b) Data point on the 1800 MHz frequency band with bad RSRP and UL

Figure 4.19: Example showing the comparison between the 800 MHz frequency band with good RSRP and the 1800 MHz frequency band with poor RSRP in the MG21’s results, demonstrating better UL performance in the former.

While analyzing the data from the tests, it was discovered that the upload speed regularly hit 0 Mbps. Each time this happened it was only for a second before the speed jumped up again. This is most likely the points where the MG21

is performing a handover. The MG21 does not support CA for UL, only for DL [65], so the MG21 can only use one eNodeB for UL at the time. This means that during the handover when the MG21 is connecting to a new eNodeB, it will be unable to use UL and the upload speed will drop to 0 Mbps.

4.6 Download speed

Since there were only two tests done with download speed, the pattern that forms is not as definitive as the one formed by the upload speed. Still, both of the tests show a similar pattern. Figure 4.20 shows the download speed for test 1 on 26.04, driving direction north in Figure 4.20a on the left and south in Figure 4.20b on the right.

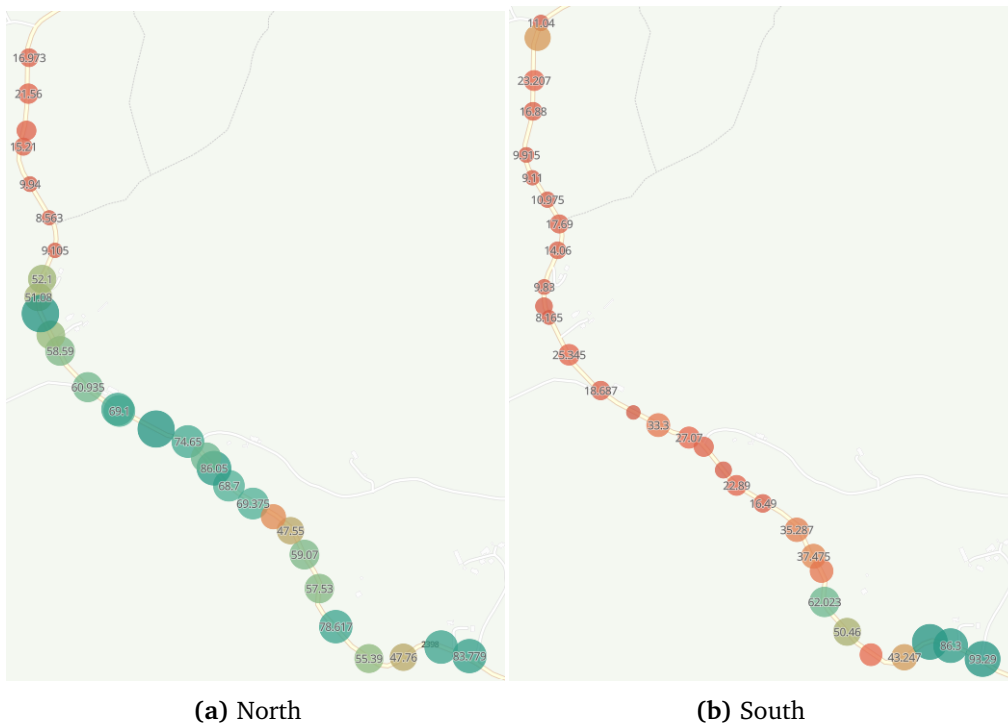


Figure 4.20: The MG21's download speed from test 1 on 26.04

Figure 4.20 shows that there is a much greater difference between the north and south here than when looking at upload speed. This can indicate that download speed is more dependent on the frequency band being used than upload speed, as mentioned in Section 4.4.2. The values for download speed are generally much higher than those for upload speed. This can also be observed when comparing Figure 4.22 with Figure 4.18. The download speed follows the frequency band much more closely in Figure 4.22 than the upload speed does in Figure 4.18. Although the data suggests this pattern, only the two tests conducted

on 26.04 have data for download speed, so there might be some variation with more data.

4.6.1 Frequency band and Bandwidth

As shown in Figure 4.21, the relationship between frequency band and download speed is about the same as for upload speed in Figure 4.17. The coverage bands with lower frequency has a lower download speed, while the capacity bands with a higher frequency has a higher download speed. As mentioned in Section 4.5.1 there are not that many data points for the 900 MHz and the 2100 MHz frequency bands, so even though they line up with theory in here, there is too little data to say much about them.

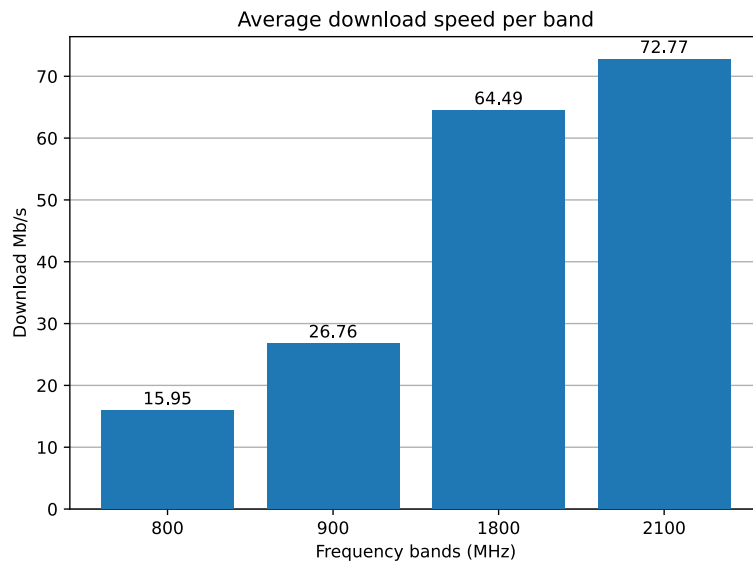


Figure 4.21: The MG21's average download speed per frequency band

Figure 4.22 also shows that download speed trends similarly to upload speed. As with upload speed, the frequency band makes the biggest difference in speed, but it is also affected by the RSRP.

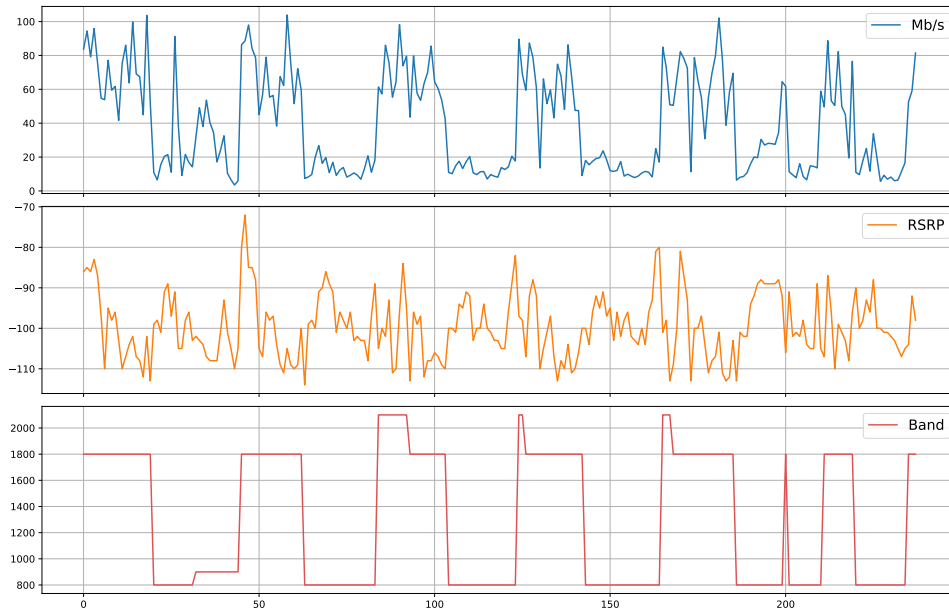


Figure 4.22: The download speed, RSRP, and frequency band of the MG21 for the tests performed on 26.04.

Unlike the upload speed, the download speed never drops to 0 Mbps. During the tests with the download speed, it only drops below 1 Mbps once, but never to 0 Mbps. This is probably because the MG21 supports CA for DL, unlike UL. This means that the MG21 can use multiple eNodeBs for DL and can therefore stay connected at all times, even during a handover. CA can also affect the results in another way. The data from the MG21 only shows the RSRP and RSRQ from the worst connection when using CA. This means that the download speed can be good while the MG21 has terrible signal parameter as it can have a good connection using CA that won't show up in the data from the API.

4.7 Results summary

To recap the findings from the results:

- The MG21 did connect to other frequency bands beside the 800 MHz and the 1800 MHz frequency band observed in the TEMS reading (Section 4.2.2).
- The MG21 was able to connect to the capacity bands more often than the smartphone was, thus indicating that the MG21 has superior antenna radiation (Section 4.2).
- Generally, the signal quality and bandwidth decreased for a short duration during handovers (Section 4.4.2).

- The driving direction did have a noticeable difference for both the MG21 and the smartphone when it comes to what frequency band they were connected to (Section 4.2.2).
- The capacity bands did not always achieve a higher bandwidth than the coverage bands, but the capacity bands had a much higher average bandwidth (Section 4.5.1).
- When the MG21's RSRP increased rapidly, it switched to a higher frequency bands. When the RSRP decreased significantly, it switched to a lower frequency bands (Section 4.4).
- After switching frequency bands, the MG21's RSRP decreased (Section 4.4).
- The RSRP and RSRQ values from both the MG21 and the smartphone followed the same trends (Section 4.3.1).
- The MG21 had overall higher RSRP values than the smartphone, but the smartphone had overall higher RSRQ values than the MG21 (Section 4.3.1).
- On the MG21, both RSRP and RSRQ was overall better on the coverage bands. On the smartphone, both RSRP and RSRQ was overall better on the capacity bands (Section 4.3.5).
- The weather did not have any noticeable effect on the signal (Sections 4.3.3).
- Heavy snow in the trees may have had a noticeable effect on the RSRQ (Sections 4.3.4 and 4.3.5).

Chapter 5

Superparameter

This chapter will explore the possibility of creating a superparameter that can be used to grade the quality of 4G in an area. For example, the superparameter can have three levels: green for good quality, yellow for medium, and red for poor quality. The red level indicates a weak connection to the 4G network but still allows some leeway before the connection is completely lost. By implementing this system, it would be possible to set up an alarm or notification that triggers whenever a device detects the superparameter dropping to the red level. This way, countermeasures can hopefully be implemented before the quality deteriorates further, reaching a complete loss of connection.

5.1 The different parameters

The results from the tests show that these four parameters are more or less interconnected, especially RSRQ, which is calculated using RSRP as one of its main components. As seen in Figures 4.18 and 4.22, bandwidth is strongly influenced by the current RSRP and frequency band. A similar pattern between RSRQ and bandwidth also exists, although not depicted in the figures. A superparameter may not require all four parameters, as two or three might be sufficient. The thresholds used to determine the level of the superparameter are based on the values from Table 5.1 for RSRP and RSRQ. Frequency bands do not necessarily have specific cutoffs, but a higher frequency band could be given more weight than a lower frequency band. However, it is challenging to determine a value for bandwidth that would classify the superparameter as being in the red level, as there are no known minimum upload and download speed requirements for the quarry use-case mentioned in Section 1.1.

Signal bar	RSRP(dBm)	RSRQ(dB)
5	RSRP \geq -83	RSRQ \geq -7
4	-83 $>$ RSRP \geq -92	-7 $>$ RSRQ \geq -10
3	-92 $>$ RSRP \geq -102	-10 $>$ RSRQ \geq -13
2	-102 $>$ RSRP \geq -111	-13 $>$ RSRQ \geq -16
1	-111 $>$ RSRP \geq -140	-16 $>$ RSRQ \geq -20

Table 5.1: Signal bar threshold for RSRP and RSRQ, specific for Cisco Meraki gateways [68]. (Reused Table 3.2)

5.1.1 RSRP and RSRQ

As mentioned above, RSRP and RSRQ are connected because RSRP is part of the calculation of RSRQ. This means that a low RSRP will always have a negative effect on the RSRQ. However, it does not imply that a low RSRP will always result in a bad RSRQ due to other factors involved in the RSRQ calculations, such as interference. For instance, even with a good RSRP in an area with significant interference, the RSRQ can still be poor. This is the main reason why, although RSRP and RSRQ are interconnected, both parameters can be used in creating a superparameter. Similarly, the thresholds for acceptable values can be based on the number of signal bars displayed by the device at different values. Table 5.1 illustrates these values for the MG21, which are likely to be considered for this superparameter. It's important to note that these values may vary across different devices, but as long as the measurements are within a reasonable range of what the MG21 obtains, it should be feasible to utilize the same threshold for different devices.

5.1.2 Bandwidth

Another parameter that can be used is bandwidth. However, due to most tests measuring upload and only two tests measuring download, some more thought has to be put into using bandwidth as a superparameter. Since upload and download speeds are usually uneven, with download often being greater than upload, caution must be exercised to prevent the difference between them from significantly affecting the superparameter. This issue becomes more apparent when considering the fact that the MG21 supports CA on download but not on upload, often resulting in much higher peaks and average speeds for download. One way to mitigate the impact of bandwidth, particularly CA, on the superparameter could be to set a maximum value for its influence. For example, any value above 30 Mbps could be capped at 30 Mbps. Determining the threshold at which bandwidth would push the superparameter into the red level is also challenging, partly due to the differences between upload and download speeds and partly due to the lack of knowledge about the minimum upload and download requirements in the quarry use-case.

5.1.3 Frequency band

Which frequency band is currently in use could affect the superparameter, as a higher frequency band has higher capacity bandwidth. This makes the current frequency band affect the current bandwidth, however the current bandwidth is also affected by RSRP and RSRQ. Frequency band could also be looked at as a double edged sword, as a lower frequency band with good RSRP and RSRQ can perform better than a higher frequency band with worse values as can be seen in Figure 4.19. Therefore it could be wise to limit how much of a positive effect a higher frequency band has on the superparameter if it were to be included in the calculation.

5.2 Why not just use bandwidth?

Mapping out 4G coverage using a speed test might seem like the easiest solution. Simply conducting a speed test and mapping the area beforehand could work well if the area remains constant and the coverage is expected to remain stable, such as in a warehouse. However, this approach won't be suitable for the quarry use-case mentioned in Section 1.1. In a quarry, the area gradually changes as the digging goes deeper and deeper, rendering the speed test method ineffective.

In this case, it would be more beneficial if the vehicles themselves could check the coverage while doing their job. But the way a speed test works is by trying to max out and use as much bandwidth as possible, and it would therefore compete with the other things that require bandwidth. Even if the speed test were given a low priority in comparison to the main task, they would still disturb one another. And when there are several vehicles doing working at once, their speed tests would interfere with each other. That might cause inaccurate results of the speed tests. Therefore it would not be a good solution to have the vehicles constantly do speed tests to check the coverage.

So the question then becomes: how to check the coverage without interfering with other applications that use bandwidth? There will now come two propositions of ways to do this, A and B, with advantages and disadvantages to each method.

5.3 Proposition A

The first proposition involves making each of the remaining parameters into a specific range of values so that they are easily comparable to each other. In this case RSRP, RSRQ and frequency band are each made into a variable between 0 and 1. Then each of them is multiplied with a weight depending on how much they matter compared to the others and then all of them are added together to form a single value.

5.3.1 The formula

Let's start with making each into a variable between 0 and 1. The way to do this is to take the value, subtract the maximum value and divide it by the maximum value subtracted from the minimum value. For RSRP the maximum value is -44 dBm and the minimum value is -140 dBm (Section 2.3.1). Because RSRP consists of negative values, add 1 so that instead of getting a value between 0 and -1, you get a value between 0 and 1. The formula then becomes:

$$\frac{RSRP - (-44)}{(-44) - (-140)} + 1 = \frac{RSRP + 44}{96} + 1 \quad (5.1)$$

For RSRQ, the maximum value is -3 dB, and the minimum value is -19.5 dB (Section 2.3.2). For frequency bands used in 4G LTE by Telenor, the maximum value is 2600 MHz, and the minimum value is 700 MHz [14, p.12-13]. Since frequency bands already consist of positive values, there is no need to add 1 to it. Additionally, the weight for RSRP is X, RSRQ is Y, and Band (frequency band) is Z. The total formula then becomes:

$$X \left(\frac{RSRP + 44}{96} + 1 \right) + Y \left(\frac{RSRQ + 3}{16.5} + 1 \right) + Z \left(\frac{Band - 700}{1900} \right) \quad (5.2)$$

The result of this formula will be a value between 0 and $(X + Y + Z)$. To ensure a more predictable result, divide it by $(X + Y + Z)$ so that it falls within the range 0 and 1, regardless of the values of X, Y or Z. The modified formula is as follows:

$$\frac{X \left(\frac{RSRP+44}{96} + 1 \right) + Y \left(\frac{RSRQ+3}{16.5} + 1 \right) + Z \left(\frac{Band-700}{1900} \right)}{X + Y + Z} \quad (5.3)$$

The advantage of this approach is that if X, Y, and Z are set to the right values, they could sufficiently mimic the upload/download speed, eliminating the need for continuous speed tests. The drawback, however, is that it may not achieve perfection. While it could approximate the speed accurately in 99.9% of cases, there is a risk of occasional significant deviations.

5.3.2 Testing the formula

To test whether the formula has potential, it will be applied to all data points from the MG21 across all nine tests conducted between 01.03 and 26.04. This means that the term 'speed' refers to either upload or download speed, depending on the specific test. Subsequently, all data points will be categorized into five groups based on the number of signal bars present on the MG21 at each point, ranging from 1 to 5 signal bars (see Table 3.2). A table displaying the data points resulting in the lowest and highest formula values for each category will be presented here for evaluation.

Ideally, the superparameter should be low when the signal bar or speed is low, and high when the signal bar or speed is high.

Test 1: $X = Y = Z = 1$

First, let's try with all the weights being equal to 1. Table 5.2 shows the results.

Signal bar	RSRP	RSRQ	Band	speed	superparameter
1	-114 dBm	-17 dB	800 MHz	7.36 Mbps	0.16
1	-113 dBm	-10 dB	2100 MHz	6.97 Mbps	0.53
2	-107 dBm	-16 dB	800 MHz	11.49 Mbps	0.2
2	-106 dBm	-7 dB	2100 MHz	23.12 Mbps	0.62
3	-101 dBm	-13 dB	800 MHz	3.96 Mbps	0.28
3	-95 dBm	-8 dB	2100 MHz	23.74 Mbps	0.63
4	-91 dBm	-10 dB	800 MHz	20.38 Mbps	0.38
4	-71 dBm	-9 dB	2100 MHz	23.83 Mbps	0.7
5	-83 dBm	-7 dB	800 MHz	12.98 Mbps	0.47
5	-80 dBm	-7 dB	800 MHz	19.04 Mbps	0.48

Table 5.2: Testing superparameter Proposition A with data from all tests, using the variables $X = Y = Z = 1$

This did not yield very good results mainly because of row two where a low signal bar and a low speed still managed to get a high value on the superparameter. Maybe if the RSRP and RSRQ mattered more than the Band, it would help.

Test 2: $X = Y = 3$ and $Z = 1$

Now let's try with X and Y being equal to 3 and Z equal to 1. Table 5.3 shows the results.

Signal bar	RSRP	RSRQ	Band	speed	superparameter
1	-114 dBm	-17 dB	800 MHz	7.36 Mbps	0.19
1	-113 dBm	-9 dB	1800 MHz	4.02 Mbps	0.48
2	-105 dBm	-16 dB	800 MHz	38.47 Mbps	0.25
2	-106 dBm	-7 dB	2100 MHz	23.12 Mbps	0.58
3	-101 dBm	-13 dB	800 MHz	3.96 Mbps	0.35
3	-95 dBm	-7 dB	1800 MHz	14.99 Mbps	0.61
4	-91 dBm	-10 dB	800 MHz	20.38 Mbps	0.47
4	-71 dBm	-9 dB	2100 MHz	23.83 Mbps	0.69
5	-83 dBm	-7 dB	800 MHz	12.98 Mbps	0.59
5	-80 dBm	-7 dB	800 MHz	19.04 Mbps	0.6

Table 5.3: Testing superparameter Proposition A with data from all test, using the variables $X = Y = 3$ and $Z = 1$

This yielded better results as it generally increased the superparameter for entries with 4-5 signal bars. However, row two still receives an excessively high value. Maybe increasing the weight on RSRP would help.

The tests continue in this manner until the results are satisfactory, and the last test, Test 3, will also be showcased here.

Test 3: $X = 7$, $Y = 2$ and $Z = 1$

Let's try with X equal to 7, Y equal to 2 and Z equal to 1. Table 5.4 shows the results.

Signal bar	RSRP	RSRQ	Band	speed	superparameter
1	-114 dBm	-17 dB	900 MHz	2.01 Mbps	0.23
1	-113 dBm	-10 dB	2100 MHz	6.97 Mbps	0.39
2	-108 dBm	-15 dB	800 MHz	20.76 Mbps	0.29
2	-72 dBm	-15 dB	1800 MHz	88.59 Mbps	0.61
3	-102 dBm	-13 dB	800 MHz	4.93 Mbps	0.36
3	-77 dBm	-11 dB	1800 MHz	30.09 Mbps	0.62
4	-92 dBm	-10 dB	800 MHz	20.34 Mbps	0.47
4	-71 dBm	-9 dB	2100 MHz	23.83 Mbps	0.7
5	-83 dBm	-7 dB	800 MHz	12.98 Mbps	0.57
5	-80 dBm	-7 dB	800 MHz	19.04 Mbps	0.59

Table 5.4: Testing superparameter Proposition A with data from all tests, using the variables $X = 7$, $Y = 2$ and $Z = 1$

Now this set of variables could work. If the limit for "bad" is set to 0.4, that would match with all of the test results. So with the limited amount of test data collected from the MG21 during this project, this is good. That being said, due to

the limitations of only using the same test area over and over, it is uncertain that this formula would work in a completely different test area with a different set of conditions.

5.4 Proposition B

A second possible solution for the superparameter could be directly based on the RSRP and RSRQ values that are considered too low. With this in mind, three variants of the same setup have been created. The first variant, or base variant, only utilizes RSRP and RSRQ when creating a superparameter. The second variant adds bandwidth as well to account for edge cases where the bandwidth is poor even though the RSRP and RSRQ are good, or vice versa. After testing the first two variants, the third variant was made by expanding upon the second variant. The key difference between the second and third variants, is the third variant's ability to weigh good and bad values separately.

5.4.1 Base variant

The formula for the base variant consists of four variables that go alongside the RSRP and RSRQ. These variables are $RSRP_{Min}$, $RSRQ_{Min}$, $RSRQ_{Multiplier}$, and Base. The Min values are the thresholds for the minimum accepted values for RSRP and RSRQ. The Base variable is used to ensure that the superparameter is greater than zero, even if the RSRP and RSRQ are at their lowest possible values. The Base variable can also serve as the threshold for when the superparameter goes into the red level. The $RSRQ_{Multiplier}$ is used to give a slightly higher weight to RSRQ compared to RSRP, as it has a smaller range of possible values.

The calculation of the formulas is fairly straightforward. It subtracts the absolute values of RSRP and RSRQ from their respective min values and multiplies the sum of RSRQ with the $RSRQ_{Multiplier}$. Finally, the two sums are added together with the Base value. Equation 5.4 shows how the formula is constructed, while Equation 5.5 provides an example with specific values assigned to the variables.

$$(RSRP_{Min} - RSRP) + RSRQ_{Multiplier} (RSRQ_{Min} - RSRQ) + Base \quad (5.4)$$

$$(101 - 103) + 3(12 - 14) + 90 = 82 \quad (5.5)$$

In the example Equation 5.5, the values for $RSRP_{Min}$ and $RSRQ_{Min}$ are derived from the values in the signal bar Table 3.2 for the MG21. A RSRP of -102 dBm or RSRQ of -13 dB represents the thresholds for 2 signal bars. The choice of 2 signal bars as the threshold is based on the need to promptly detect unstable or

weak connections in the area. However, one of the strengths of this proposition is the ability to freely adjust these variables, especially if an alternative value yields better overall results.

5.4.2 Variant with bandwidth

The second variant uses the same calculation formula as the base variant but also incorporates bandwidth into the calculation of the superparameter. This is because upload and download speeds can vary with slight changes in RSRP and RSRQ, and the current frequency band can have a significant impact as well. Another issue is that the MG21 supports CA for download but not for upload. Considering these factors, the formula including bandwidth introduces two additional variables: $Speed_{Min}$ and $Speed_{Max}$. $Speed_{Min}$ functions similarly to $RSRP_{Min}$ and $RSRQ_{Min}$, setting the threshold at which the bandwidth is negatively counted towards the superparameter. $Speed_{Max}$ is used to limit the maximum positive impact of the bandwidth on the superparameter. This limitation is particularly important to restrict the influence of CA on the parameter since RSRP and RSRQ cannot be observed for cells other than the serving cell of the MG21. This involves the use of an IF statement, indicating that all bandwidth values above $Speed_{Max}$ should be considered as $Speed_{Max}$ in value.

Pseudocode for the second variant:

```

RSRPSum = MinRSRP - RSRP
RSRQSum = RSRQMultiplier*(MinRSRQ-RSRQ)
IF (Speed > MaxSpeed) {
    Speed = MaxSpeed
}
SpeedSum = Speed - MinSpeed
Superparameter = RSRPSum + RSRQSum + SpeedSum + Base

```

One of the challenges in adopting this approach to include bandwidth is deciding the values of the $Speed_{Min}$ and $Speed_{Max}$ variables. While you want the parameter to have a noticeable effect on the superparameter, you do not want it to overshadow the other variables completely. Additionally, there is the issue of not knowing the bandwidth requirements for the automated tip-trucks in the quarry use-case.

5.4.3 Testing the first and second variants

The first two variants were tested using modified versions of the script used to test Proposition A (Section 5.3). Separate versions of the script were made for each variant. The main change to the script for the first variant was to replace the formula used in the base variant (Formula 5.4) and modify some variable names to align with this formula. The second variant built upon the script for the first variant by adding bandwidth to the score calculation.

When testing the variants, it is possible to adjust all the variables to determine if they yield better results for the superparameter. Based on the values assigned to these variables, it can be decided whether the Base variable will be used for the green, yellow, and red thresholds of the superparameter, or if the thresholds should be based on something else.

First variant test

For this test of the first variant, the minimum values for RSRP and RSRQ were set to -101 dBm and -12 dB, respectively. These values were chosen because they are on the border when the MG21 goes from 3 down to 2 signal bars (Table 5.1). The $RSRQ_{Multiplier}$ is set to 3 to give more weight to single digit changes in RSRQ compared to single digit changes in RSRP. The base value of 90 is chosen based on the superparameter, ensuring it does not go into negative values even with the worst RSRP and RSRQ values possible. The result of this test using all MG21 data from all the tests can be seen in Table 5.5.

Signal bar	RSRP	RSRQ	speed	superparameter
1	-114 dBm	-17 dB	2.01 Mbps	62
1	-112 dBm	-9 dB	1.01 Mbps	88
2	-111 dBm	-16 dB	55.35 Mbps	68
2	-72 dBm	-15 dB	88.59 Mbps	110
3	-102 dBm	-13 dB	7.99 Mbps	86
3	-77 dBm	-11 dB	30.09 Mbps	117
4	-92 dBm	-10 dB	26.2 Mbps	105
4	-71 dBm	-9 dB	23.83 Mbps	129
5	-83 dBm	-7 dB	12.98 Mbps	123
5	-80 dBm	-7 dB	19.04 Mbps	126

Table 5.5: Example of a superparameter test with data from all tests, using the first variant, with $RSRP_{Min} = -101$, $RSRQ_{Min} = -12$, $RSRQ_{Multiplier} = 3$, Base = 90

Table 5.5 clearly shows the main flaw of this variant. The greater the superparameter, the better the result. However, the best results with a lower signal bar

are almost always better than the lowest results with the next higher signal bar. In this table, the best result with 1 signal bar is better than the worst result with 3 signal bars. So while the superparameter gradually improves as the number of signal bars increases, it is not a reliable indicator of low connection quality. If the base variable is used as the threshold for the superparameter, stating that all values below 90 are considered in the red level, then all measurements with 1 signal bar would be in the red level. The lower end of 2 signal bars would also be in the red level. However, the higher end of 2 signal bars is much better than 90 and could possibly be in the green level, depending on where the threshold between green and yellow is set. At the same time, the lower end of 3 signal bars is also below the base value and would be considered in the red level. Using the RSRP and RSRQ values on the border between 2 or 3 signal bars would ideally categorize all values with 2 signal bars in the red level, while all values with 3 signal bars should be in the yellow level. Another issue is that the best value with 4 signal bars is better than the best value with 5 signal bars.

Changing the value for the $RSRQ_{Multiplier}$ alone does not make much of a difference; instead, it primarily results in lower superparameter values becoming even lower and higher ones becoming higher. This leads to increased overlaps between the highest values on low signal bars and the lowest values on higher signal bars.

Second variant test

The second variant was tested using many of the same values for the variables as the first variant. Additionally, new variables for $Speed_{Min}$ and $Speed_{Max}$ were added. These were set to 5 and 15 Mbps, respectively. Bandwidth values below 5 Mbps were considered negative, and any speed above 15 Mbps was capped at 15 Mbps.

The $Speed_{Min}$ value is not based on anything relative to the quarry use-case, as the minimum bandwidth requirements for the automated tip-trucks are unknown. Therefore, the $Speed_{Min}$ is set to 5 Mbps, loosely based on the minimum recommended download speed for full HD video streaming [72]. While the tip-trucks probably won't need to stream full HD video, they might require a similar amount of bandwidth. The tip-trucks are likely sending a live video feed to an operator, together with live updates for their position and status. They are also required to download mapping updates from The Norwegian Mapping Authority [73].

Signal bar	RSRP	RSRQ	speed	superparameter
1	-114 dBm	-17 dB	2.01 Mbps	59.01
1	-112 dBm	-11 dB	9.08 Mbps	86.08
2	-110 dBm	-14 dB	1 Mbps	71
2	-72 dBm	-15 dB	88.59 Mbps	120
3	-102 dBm	-12 dB	1 Mbps	85
3	-77 dBm	-11 dB	30.09 Mbps	127
4	-91 dBm	-10 dB	11.35 Mbps	112.35
4	-71 dBm	-9 dB	23.83 Mbps	139
5	-83 dBm	-7 dB	12.98 Mbps	130.98
5	-80 dBm	-7 dB	19.04 Mbps	136

Table 5.6: Example of a superparameter test with data from all tests, using the second variant, with $RSRP_{Min} = -101$, $RSRQ_{Min} = -12$, $RSRQ_{Multiplier} = 3$, $Speed_{Min} = 5$, $Speed_{Max} = 15$, Base = 90

The superparameter result from the second variant in Table 5.6 shows a similar pattern to the first variant. Some of the measurements used for the best and worst signal bar on a specific signal have changed with the introduction of bandwidth. However, the best results on a low signal bar still show better superparameter values than the lowest values on a higher signal bar count. This variant addresses edge cases where there is low bandwidth even when both RSRP and RSRQ are good, as well as high bandwidth when both RSRP and RSRQ are low. Nevertheless, the best value on four signal bars is better than the best value on five signal bars. Overall, the results are quite similar to the first variant.

5.4.4 Variant with separate weighting

During testing of the first and second variants, the results showed a trend: the highest values that would result in a low number of signal bars on the MG21 had a higher score on the superparameter compared to the lowest values with higher numbers of signal bars. One idea to counteract this trend was to introduce the possibility of assigning separate weights to values below the minimum accepted values for RSRP, RSRQ, and bandwidth. This approach was specifically employed to ensure that low values would have a greater negative impact on the superparameter than the positive impact of high values. For instance, a negative RSRP could be weighted ten times more heavily than a positive value. Additionally, this approach would facilitate a faster decline of the superparameter to a red level when any parameter drops too low, unlike the first and second variants where multiple parameters generally had to fall below their respective minimum values to trigger the superparameter to reach a red level.

The new weights added to negative values are added as three new variables: $negRSRP_{Multiplier}$, $negRSRQ_{Multiplier}$, and $negSpeed_{Multiplier}$. These variables are

incorporated into the calculation of the superparameter score by adding IF statements to modify how the sub-calculations involving them are performed.

Pseudocode for the third variant:

```

if rsrp < minRsrp : rsrpSum = minRsrp-rsrp
else : rsrpSum = negRsrpMultiplier*(minRsrp-rsrp)

if rsrq < minRsrq : rsrqSum = rsrqMultiplier*(minRsrq-rsrq)
else : rsrqSum = negRsrqMultiplier*(rsrqMultiplier*(minRsrq-rsrq))

if speed > maxSpeed : speedSum = maxSpeed - minSpeed
elif speed < minSpeed : speedSum = negSpeedMultiplier*(speed-minSpeed)
else : speedSum = speed - minSpeed

Score = rsrpSum + rsrqSum + speedSum + Base

```

Third variant test

The third variant is tested using another modified version of the script used in the other tests. The same minimum values for RSRP, RSRQ, $Speed_{Min}$ and $Speed_{Max}$ are used. The other variables are either changed or new, with $RSRQ_{Multiplier}$ being set to 4, Base being set to 1000 and the new negative value multipliers $negRSRP_{Multiplier}$, $negRSRQ_{Multiplier}$ and $negSpeed_{Multiplier}$ being set to 10, 20 and 15 respectively. This high Base value leads to the superparameter values being about ten times higher than in the tests of the previous variants.

Signal bar	RSRP	RSRQ	speed	superparameter
1	-114 dBm	-17 dB	2.01 Mbps	424.15
1	-112 dBm	-11 dB	9.08 Mbps	898.08
2	-110 dBm	-14 dB	1 Mbps	590
2	-72 dBm	-15 dB	88.59 Mbps	1006
3	-102 dBm	-12 dB	1 Mbps	904.4
3	-77 dBm	-11 dB	30.09 Mbps	1038
4	-91 dBm	-10 dB	11.35 Mbps	1024.35
4	-71 dBm	-9 dB	23.83 Mbps	1052
5	-83 dBm	-7 dB	12.98 Mbps	1045.98
5	-80 dBm	-7 dB	19.04 Mbps	1051

Table 5.7: Example of a superparameter test with data from all tests, using the third variant, with $RSRP_{Min} = -101$, $negRSRP_{Multiplier} = 10$, $RSRQ_{Min} = -12$, $RSRQ_{Multiplier} = 4$, $negRSRQ_{Multiplier} = 20$, $Speed_{Min} = 5$, $Speed_{Max} = 15$, $negSpeed_{Multiplier} = 15$, Base = 1000

The results for the third variant in Table 5.7 are once again similar to those in Tables 5.5 and 5.6 for the first and second variants. However, the third variant

performs slightly better in terms of signal bars. While the first two variants had values overlapping two signal bars up, this is no longer the case with the third variant, where the overlap is limited to one signal bar. Furthermore, the best value on four signal bars is still better than the best value on five signal bars, although it is not yet ideal. However, this time, the values are within a single digit of each other.

If the Base variable were used as the threshold for determining when the superparameter enters the red level, this third variant would be the closest to having all values on the 2 signal bars of the MG21 in the red level. However, there is one small issue with the third variant compared to the first two. The added complexity in the code significantly increases the processing time, requiring multiple times more processing compared to the others. While this is not a problem in a non-fully automated system, it could become an issue in a fully automated system where the superparameter needs to be calculated in close to real-time.

5.5 Proposition A vs Proposition B

This section will compare the two propositions against each other, particularly their strengths and weaknesses. Proposition B has three different variants; however, this section will focus on the third variant, which includes separate weights for negative values. Note that neither proposition is perfect. More work is needed to find the ideal weights and values for the variables used.

One of the significant differences between the two propositions is that Proposition A does not require an internet speed test to run simultaneously, while Proposition B requires either download or upload speed in its calculation. Additionally, Proposition A is less complex and more predictable in terms of calculation time. This suggests that Proposition A is the better option if the measuring device has limited processing capacity. Furthermore, if real-time bandwidth measurements are out of the question, Proposition A is the only option.

When it comes to determining which of the propositions is better at assigning appropriate values to the measurements, Proposition B probably has greater potential. This is because Proposition B incorporates speed as a part of its calculation, along with separate weights for positive and negative values. This allows individual parameters to have a significant impact on the final result. Ideally, if any of the parameters drop to an unacceptable value, the superparameter should also decrease to the red level, which Proposition B accomplishes more effectively. Alternatively, a simpler solution would involve monitoring each of the parameters individually and issuing a warning whenever any of them becomes too low.

Chapter 6

Conclusion

This final chapter will draw conclusions based on the results and discussion. First comes a section that goes over the objectives defined in Section 1.2.1. Secondly, there will be a section about the testing and the resulting heatmaps. Then, a section will present conclusions regarding the superparameter. Finally, there will be some remarks and reflections on whether the MG21 is actually suitable for conducting these kinds of tests, followed by a section on future work that can be done to build upon this project.

6.1 Objectives

A test setup capable of completing the test for the project was created. It didn't turn out exactly as imagined, but performed well enough for the purpose of the project. The test data were neatly presented in a heatmap, just as planned. Several propositions for potential superparameter was pitched, but a complete superparameter was not reached. All of the propositions would require further work to be good enough for industrial use. The playbook was created exactly as it was explained in Section 3.8.

6.2 Testing and heatmaps

This section will be about the testing and the data gathered from said tests.

6.2.1 Choice of test area

Throughout all the tests, the same test area was utilized. The choice of the test area seemed to be appropriate for imitating the inconsistency of signal quality that may be present in the use-case presented in Section 1.1. The team gathered a lot of diverse signal data, which was useful when attempting to determine the superparameter. The only downside to the test area was that it was occasionally

challenging to navigate due to the weather conditions. However, this did not significantly impact the test results.

6.2.2 Lack of data

The data gathered from both test devices provided useful information for determining the signal quality throughout the test area. However, it would have been more helpful if the MG21 could have collected data other than just RSRP and RSRQ to assess the signal quality. There were some differing test results that could not be explained due to the lack of data on the interference level present in the environment and other signal quality measurements. Such data could have provided a clearer view of why the results occurred as they did.

Having access to a speed test on the smartphone would have been highly desirable as it would have allowed for a comparison of network quality between both test devices. Additionally, the smartphone provided data on parameters like CA, connected eNodeBs and cells, and other signal parameters that were not measured by the MG21. This made it more challenging to compare the results from both test devices since they were unable to gather the same data.

There were several issues related to the MG21's data gathering, primarily the problem with the Meraki API not providing sufficient data when there was a 45-second delay between each data update. This resulted in drastic changes in the values of the signal parameters between each update, making it harder to analyze how the different signal parameters behaved at all times.

6.2.3 Heatmap

The heatmap for the project was made using Elasticsearch and Kibana. While it had some minor issues for our use-case, it worked well overall. With data views in Kibana, the heatmap would automatically update whenever new data was added to Elasticsearch after each test. This, combined with a Python library that allowed data to be pushed to Elasticsearch using a script for a mostly automated process that fit well for this project. The main drawback of using Kibana for this use-case is that the heatmap does not support negative values when creating a heatmap with custom colors. However, this was a minor issue with an easy workaround. In conclusion, Elasticsearch with Kibana proved to be a good heatmap solution for this project.

6.3 The superparameter

To draw a conclusion on the superparameter is difficult due to uncertainty about what would be preferable in each case. Therefore, the conclusion here is that what is the best proposition presented in Chapter 5 depends mainly on whether

it is acceptable for each vehicle to constantly run a speed test and max out the bandwidth or not.

It is also difficult to set specific limits on what superparameter values are too low due to uncertainty about the demands of the jobs performed by the autonomous vehicles. For example, how much do they need to be able to upload and download per second? In a use-case based on the MG21, the signal bar values from Table 3.2 could be used. Minimum accepted values for RSRP and RSRQ could, for example, be set to the best value on 2 signal bars. These values could also be applied to other use-cases, but they might yield inconsistent results across different devices.

However, sometimes the simpler solution is the best, and this case is probably one of them. Each of the parameters measures something important. While the parameters affect each other, especially with RSRQ being based on RSRP and interference, bandwidth is often reflected in the other parameters as well. However, one of the values can still be significantly better or worse than the others would indicate. In some cases, RSRP is very low while RSRQ and bandwidth are still good, or vice versa. Therefore, it would be best to monitor each of them individually and react if any single one of them becomes too low. This approach would be more accurate than any of the propositions for the superparameter presented in Chapter 5. It would also reduce the required overhead of calculating the superparameter for every single measurement.

6.4 Suitability of the MG21

With the long delay between each Meraki API request, it will be hard to use the MG21 to obtain an accurate measurement of the signal quality of a cellular network. It requires that the vehicles with the MG21 device installed to drive at a relatively slow speed for the signal measurements to be useful for monitoring. One can argue that the vehicles in the quarry use-case drive at slow speed, so that the 45 second API delay might not matter as much. Therefore, the MG21 could be a good solution to measure the signal quality. However, it is important to have updated data at all times because the signal quality may change drastically within seconds, leading to the vehicles stopping before vehicle operator receives a warning.

Meraki API also lacks the ability to gather data about which frequency band the MG21 is using. The only way to gather this data from the MG21 is to manually check the Meraki dashboard, which is not an effective or automated method.

In general, the MG21 is not a suitable device for measuring the signal quality of a cellular network when it is constantly moving. For the MG21 to be suitable for gathering signal measurements while constantly moving, Cisco Meraki needs

to implement functionalities in the Meraki API. This primarily involved ensuring that API updates occur more rapidly and consistently. Additionally, the ability to see relevant signal data other than RSRP and RSRQ should be added.

6.5 Further work

There are a few things that could have been done to get a better result. The whole process of using the collected data to make heatmaps could have been fully automated and made the heatmaps in Kibana while the test was still going on. That means that scripts would have to collect all the data into a single log file with the same format. Without the frequency band for the MG21 from the Meraki API, there would still be a need to add that data to the logs manually after the test. A way to measure the bandwidth on the smartphone would also have been nice to make a better comparison with the MG21. A possibility was to develop an app designed for this exact project that gathers all the data needed in a perfect log file, to compare with the data from the MG21. Another possibility for the smartphone could have been to use TEMS, like Telenor used in their measurements of the area.

To make this project more ideal, significant changes need to be implemented, which may compromise the specifications given in the assignment (see Appendix A). Firstly, replacing the MG21 with a device designed for mobility would enhance the reliability of network measurements and data accuracy. Ideally, this device should provide the following data through the API: RSRP, RSRQ, frequency band, position data, CA data, bandwidth, ID of the base station used, and interference level. For example, a smartphone with a dedicated application for gathering this data could be used. With such a device in place, the data processing could be fully automated, and heatmaps could be generated in Kibana during the test. Consequently, there would no longer be a need for iPerf3 or an additional device.

Further work on the superparameters is needed to make good use of them. Finding the optimal set of variables has not been prioritized in this project; instead, the potential of these variables has been discussed. Another aspect to ensure the quality of the parameters would be to have more diverse test data. This could be achieved simply by testing in additional areas. With more diverse test data, one would be able to enhance both proposed superparameters, making them more robust and capable of handling different values and signal conditions effectively.

Bibliography

- [1] 'Imrad-structure - ntnu.' (May 2023), [Online]. Available: <https://i.ntnu.no/academic-writing/imrad-structure> (visited on 18/05/2023).
- [2] 'What is Bandwidth - Definition, Meaning & Explanation.' (Feb. 2023), [Online]. Available: <https://www.verizon.com/articles/internet-essentials/bandwidth-definition/> (visited on 13/04/2023).
- [3] S. Hill, S. Chandler and P. Beaton. 'LTE vs. 4G | The Differences Explained.' (Oct. 2021), [Online]. Available: <https://www.digitaltrends.com/mobile/4g-vs-lte/> (visited on 24/01/2023).
- [4] T. Weber. 'What Is 4G? | IoT Glossary.' (Apr. 2021), [Online]. Available: <https://www.emnify.com/iot-glossary/4g> (visited on 24/01/2023).
- [5] K. Sookocheff. 'How Does LTE Work?' (Jul. 2019), [Online]. Available: <https://sookocheff.com/post/networking/how-does-lte-work/> (visited on 24/01/2023).
- [6] 'Overview of LTE eNodeB functions.' (May 2022), [Online]. Available: <https://forum.huawei.com/enterprise/en/overview-of-lte-enodeb-functions/thread/880277-100305> (visited on 22/03/2023).
- [7] A. Ghayas. 'What are cells in mobile communications?' (Sep. 2019), [Online]. Available: <https://commsbrief.com/what-are-cells-in-mobile-communications/> (visited on 22/03/2023).
- [8] C. Manning. 'What are radio waves?' (Aug. 2018), [Online]. Available: http://www.nasa.gov/directorates/heo/scan/communications/outreach/funfacts/what_are_radio_waves (visited on 21/03/2023).
- [9] A. Pompos. 'Fermilab | Science | Inquiring Minds | Questions About Physics.' (Apr. 2014), [Online]. Available: <https://www.fnal.gov/pub/science/inquiring/questions/mikep.html> (visited on 11/04/2023).
- [10] 'Wavelength | Center for Science Education.' (2018), [Online]. Available: <https://scied.ucar.edu/learning-zone/atmosphere/wavelength> (visited on 11/04/2023).
- [11] 'Extreme high-frequency signals enable terabits-per-second data links.' (Mar. 2020), [Online]. Available: <https://www.sciencedaily.com/releases/2020/03/200331130029.htm> (visited on 11/04/2023).

- [12] R. Whyte. 'Multipath Interference in Indirect Time-of-Flight Depth Sensors.' (Mar. 2021), [Online]. Available: <https://medium.com/chronoptics-time-of-flight/multipath-interference-in-indirect-time-of-flight-depth-sensors-7f59c5fcd122> (visited on 21/03/2023).
- [13] S. Petryschuk. 'What is Network Congestion? Common Causes and How to Fix Them.' (May 2021), [Online]. Available: <https://www.auvik.com/franklyit/blog/what-is-network-congestion/> (visited on 13/04/2023).
- [14] 'Frekvenskompass for mobilkommunikasjon 6. utgave - november 2022,' Nov. 2022, p. 54. [Online]. Available: <https://nkom.no/aktuelt/ny-utgave-av-frekvenskompasset-for-mobil-frem-mot-2030> (visited on 12/03/2023).
- [15] K. Dennean and R. Gilligan. '5G coverage vs. capacity.' (Mar. 2020), [Online]. Available: <https://www.ubs.com/global/en/wealth-management/insights/chief-investment-office/house-view/daily/2020/5gcoverage-capacity.html> (visited on 12/03/2023).
- [16] 'What Is Low, Mid, and High-Band? The 5G Spectrum Layers Explained.' (May 2022), [Online]. Available: <https://www.westbase.io/blog/what-is-low-mid-and-high-band-the-5g-spectrum-layers-explained> (visited on 12/03/2023).
- [17] '3GPP - The Mobile Broadband Standard.' (), [Online]. Available: <https://www.3gpp.org/> (visited on 11/04/2023).
- [18] 'Frekvensauksjoner.' (), [Online]. Available: <https://www.nkom.no/frekvenser-og-elektronisk-utstyr/frekvensauksjoner> (visited on 24/03/2023).
- [19] A. Grami, 'Communication Networks,' in *Introduction to Digital Communications*. Elsevier, 2016, pp. 457–491, ISBN: 978-0-12-407682-2. DOI: 10.1016/B978-0-12-407682-2.00011-9. [Online]. Available: <https://linkinghub.elsevier.com/retrieve/pii/B9780124076822000119> (visited on 24/03/2023).
- [20] M. Rouse. 'Guard Band.' (Dec. 2011), [Online]. Available: <https://www.techopedia.com/definition/7494/guard-band-telecommunications> (visited on 24/03/2023).
- [21] 'B20(800MHz).' (), [Online]. Available: <https://halberdbastion.com/technology/cellular/4g-lte/lte-frequency-bands/b20-800-mhz> (visited on 24/03/2023).
- [22] 'B3(1800MHz).' (), [Online]. Available: <https://halberdbastion.com/technology/cellular/4g-lte/lte-frequency-bands/b3-1800-mhz> (visited on 24/03/2023).
- [23] M. L. Rocca. 'RSRP and RSRQ Measurement in LTE.' (Apr. 2016), [Online]. Available: <https://arimas.com/2016/04/04/78-rsrp-and-rsrq-measurement-in-lte/> (visited on 24/03/2023).

- [24] 'LTE RSSI, RSRP and RSRQ Measurement.' (), [Online]. Available: <https://www.cablefree.net/wirelesstechnology/4glte/rsrp-rsrq-measurement-lte/> (visited on 24/03/2023).
- [25] M. Rouse. 'Handover.' (Oct. 2013), [Online]. Available: <https://www.techopedia.com/definition/19726/handover-ho> (visited on 26/03/2023).
- [26] R. Ahmad, E. A. Sundararajan, N. E. Othman and M. Ismail, 'Handover in LTE-advanced wireless networks: state of art and survey of decision algorithm,' vol. 66, no. 3, pp. 533–558, Nov. 2017, ISSN: 1572-9451. DOI: 10.1007/s11235-017-0303-6. [Online]. Available: <https://doi.org/10.1007/s11235-017-0303-6> (visited on 26/03/2023).
- [27] 'What is Handover Interruption Time?' (Apr. 2022), [Online]. Available: <https://www.everythingrf.com/community/what-is-handover-interruption-time> (visited on 26/03/2023).
- [28] (Mar. 2019), [Online]. Available: https://www.electronicshub.org/types-of-antennas/#So_what_is_an_Antenna (visited on 26/03/2023).
- [29] 'Antenna Patterns and Their Meaning,' Jul. 2007. [Online]. Available: <https://www.industrialnetworking.com/pdf/Antenna-Patterns.pdf> (visited on 26/03/2023).
- [30] 'Cartesian coordinate system.' Page Version ID: 1145217094. (Mar. 2023), [Online]. Available: https://en.wikipedia.org/wiki/Cartesian_coordinate_system#Three_dimensions (visited on 26/03/2023).
- [31] 'What is an Isotropic Antenna?' (Oct. 2021), [Online]. Available: <https://www.everythingrf.com/community/what-is-an-isotropic-antenna> (visited on 26/03/2023).
- [32] 'What is Antenna Gain? Is Higher Antenna Gain Better?' (), [Online]. Available: <https://www.inhandnetworks.com/article/277.html> (visited on 26/03/2023).
- [33] P. J. Bevelacqua. 'Antenna Gain.' (), [Online]. Available: <https://www.antenna-theory.com/basics/gain.php> (visited on 26/03/2023).
- [34] 'Omnidirectional antenna.' Page Version ID: 1145894646. (Mar. 2023), [Online]. Available: https://en.wikipedia.org/w/index.php?title=Omnidirectional_antenna&oldid=1145894646 (visited on 29/03/2023).
- [35] 'Antenna Radiation Patterns: Why They're Important.' (Feb. 2021), [Online]. Available: https://www.netxl.com/blog/antennas/radiation-patterns/#heading_3 (visited on 29/03/2023).
- [36] 'The difference between Directional and Omni-Directional Antennas.' (Sep. 2019), [Online]. Available: <https://www.netxl.com/blog/networking/directional-or-omni-directional-antennas> (visited on 29/03/2023).

- [37] L. Miller, 'Carrier Aggregation Fundamentals For Dummies,' p. 23, 2016. [Online]. Available: <http://cdn.everythingrf.com/live/qorvo-carrier-aggregation-fundamentals-for-dummies-volume-1.pdf> (visited on 29/03/2023).
- [38] G. D. Ntouni, A.-A. A. Boulogeorgos, D. S. Karas, T. A. Tsiftsis, F. Foukalas, V. M. Kapinas and G. K. Karagiannidis, 'Inter-band carrier aggregation in heterogeneous networks: Design and assessment,' Aug. 2014, pp. 842–847. DOI: 10.1109/ISWCS.2014.6933470. (visited on 29/03/2023).
- [39] J. Wannstrom. 'Carrier Aggregation explained.' (Dec. 2022), [Online]. Available: <https://www.3gpp.org/technologies/101-carrier-aggregation-explained> (visited on 29/03/2023).
- [40] L. Ayvazyan Armen. Husikyan. 'LTE Carrier Aggregation and Applicability IntereNodeB Carrier Aggregation.' (), [Online]. Available: <https://csit.am/2017/Proceedings/TN/TNp1.pdf> (visited on 10/05/2023).
- [41] 'Cellular Modem vs Cellular Router vs Cellular Gateway: What's the difference.' (), [Online]. Available: <https://www.inhandnetworks.com/modem-router-gateway.html> (visited on 16/04/2023).
- [42] 'MG21 Datasheet.' (), [Online]. Available: <https://meraki.cisco.com/product-collateral/mg21-datasheet/?file> (visited on 20/05/2023).
- [43] 'About Cisco.' (May 2023), [Online]. Available: <https://www.cisco.com/c/en/us/about.html> (visited on 20/05/2023).
- [44] 'Cisco meraki.' en. Page Version ID: 1146472173. (Mar. 2023), [Online]. Available: https://en.wikipedia.org/w/index.php?title=Cisco_Meraki&oldid=1146472173 (visited on 10/05/2023).
- [45] 'Mg cellular gateways.' (), [Online]. Available: <https://meraki.cisco.com/product-collateral/mg-cellular-gateways/?file> (visited on 16/04/2023).
- [46] M. Team. 'Mg21 roaming setup,' community.meraki.com. (Feb. 2020), [Online]. Available: <https://community.meraki.com/t5/Wireless-WAN/MG21-roaming-setup/m-p/75602> (visited on 16/04/2023).
- [47] J. Clark. 'What is the Internet of Things, and how does it work?' (Nov. 2016), [Online]. Available: <https://www.ibm.com/blogs/internet-of-things/what-is-the-iot/> (visited on 30/03/2023).
- [48] 'Nordic Thingy 91.' (), [Online]. Available: <https://www.nordicsemi.com/Products/Development-hardware/Nordic-Thingy-91> (visited on 30/03/2023).
- [49] 'Unraid.' (May 2023), [Online]. Available: <https://unraid.net/product> (visited on 20/05/2023).

- [50] 'Meraki Cloud Architecture.' (May 2023), [Online]. Available: https://documentation.meraki.com/Architectures_and_Best_Practices/Cisco_Meraki_Best_Practice_Design/Meraki_Cloud_Architecture (visited on 20/05/2023).
- [51] U. Rajagopalan, 'MG41/41E Datasheet,' p. 17. [Online]. Available: https://www.t-mobile.com/content/dam/tfb/pdf/tfb-iot/MG41_41E_Datasheet.pdf.
- [52] 'Cisco Meraki Dashboard API.' (May 2023), [Online]. Available: https://documentation.meraki.com/General_Administration/Other_Topics/Cisco_Meraki_Dashboard_API (visited on 21/05/2023).
- [53] 'JSON Format.' (), [Online]. Available: <https://www.json.org/json-en.html> (visited on 20/05/2023).
- [54] 'About Speedtest.' (May 2023), [Online]. Available: <https://www.speedtest.net/about> (visited on 21/05/2023).
- [55] 'iperf3: A TCP, UDP, and SCTP network bandwidth measurement tool.' (May 2023), [Online]. Available: <https://github.com/esnet/iperf> (visited on 21/05/2023).
- [56] 'GPSLogger.' (May 2023), [Online]. Available: <https://github.com/mendhak/gpslogger> (visited on 21/05/2023).
- [57] 'G-MoN Pro - Apps on Google Play.' (May 2023), [Online]. Available: https://play.google.com/store/apps/details?id=de.carknue.gmonpro&hl=en_US (visited on 21/05/2023).
- [58] 'What is Microsoft Teams? - Microsoft Support.' (May 2023), [Online]. Available: <https://support.microsoft.com/en-us/topic/what-is-microsoft-teams-3de4d369-0167-8def-b93b-0eb5286d7a29> (visited on 21/05/2023).
- [59] 'Download PuTTY - a free SSH and telnet client for Windows.' (May 2023), [Online]. Available: <https://support.microsoft.com/en-us/topic/what-is-microsoft-teams-3de4d369-0167-8def-b93b-0eb5286d7a29> (visited on 21/05/2023).
- [60] 'Elastic Stack: Elasticsearch, Kibana, Beats & Logstash.' (May 2023), [Online]. Available: <https://www.elastic.co/elastic-stack> (visited on 21/05/2023).
- [61] 'Elastic Stack. Meet the search platform that helps you search, solve, and succeed.' (), [Online]. Available: <https://www.elastic.co/elastic-stack/> (visited on 16/04/2023).
- [62] 'What is Python? Executive Summary.' (May 2023), [Online]. Available: <https://www.python.org/doc/essays/blurb/> (visited on 21/05/2023).
- [63] 'Wiki - Wiki | OBS.' (May 2023), [Online]. Available: <https://obsproject.com/wiki/> (visited on 21/05/2023).

- [64] 'MG21 Internal Antenna Datasheet 2020.' (Oct. 2020), [Online]. Available: https://documentation.meraki.com/MG/MG21_Internal_Antenna_Datasheet (visited on 27/03/2023).
- [65] 'MG21/21E Technical Specifications.' (), [Online]. Available: https://documentation.meraki.com/MG/MG_Datasheets/MG21%2F%2F21E_Datasheet.
- [66] 'Mobile Main and AUX Antennas.' (), [Online]. Available: https://wiki.teltonika-networks.com/view/Mobile_Main_and_AUX_Antennas (visited on 11/04/2023).
- [67] P. J. Bevelacqua. 'PIFA - Planar Inverted-F Antennas.' (), [Online]. Available: <https://www.antenna-theory.com/antennas/patches/pifa.php> (visited on 11/04/2023).
- [68] 'MG21 Troubleshooting Guide 2022.' (Aug. 2022), [Online]. Available: https://documentation.meraki.com/MG/MG_Troubleshooting/MG21_Troubleshooting_Guide (visited on 24/03/2023).
- [69] 'TEMS Suite – Mobile Network Testing | Infovista.' (May 2023), [Online]. Available: <https://www.infovista.com/products/network-testing-optimization/tems-suite> (visited on 21/05/2023).
- [70] 'B8(900MHz).' (), [Online]. Available: <https://halberdbastion.com/technology/cellular/4g-lte/lte-frequency-bands/b8-900-mhz> (visited on 06/05/2023).
- [71] 'B1(2100MHz).' (), [Online]. Available: <https://halberdbastion.com/technology/cellular/4g-lte/lte-frequency-bands/b1-2100-mhz> (visited on 06/05/2023).
- [72] L. Walker. 'What is a good broadband speed for streaming?' Comparethemarket. (Apr. 2023), [Online]. Available: <https://www.comparethemarket.com/broadband/content/broadband-for-streaming/> (visited on 15/05/2023).
- [73] T. N. M. Authority's. 'From outer space to the deepest ocean floors.' (Jan. 2021), [Online]. Available: <https://www.kartverket.no/en/about-kartverket/what-we-do> (visited on 20/05/2023).

Appendix A

Assignment

This appendix contains the project assignment given by Telenor.

Driver i markedet

Telenor ser at industrien trenger nye mobile løsninger og at det vil være av kritisk karakter å avdekke kvalitets utfordringer i mobilnettet som påvirker de industrielle use-casene

<https://www.telenor.no/bedrift/aktuelt/5g/industri/>

Entreprenøren Romarheim og teknologileverandøren Steer hadde behov for spesialtilpassing av nettløsningen sin i et steinbrudd på Eikefet. I rutene TipTruckene kjørte, måtte Telenor sørge for en kontinuerlig god mobildekning. Dersom TipTruckene mister 4G nettet, går den i «safe mode» og stopper opp sammen med produksjonen i steinbruddet.



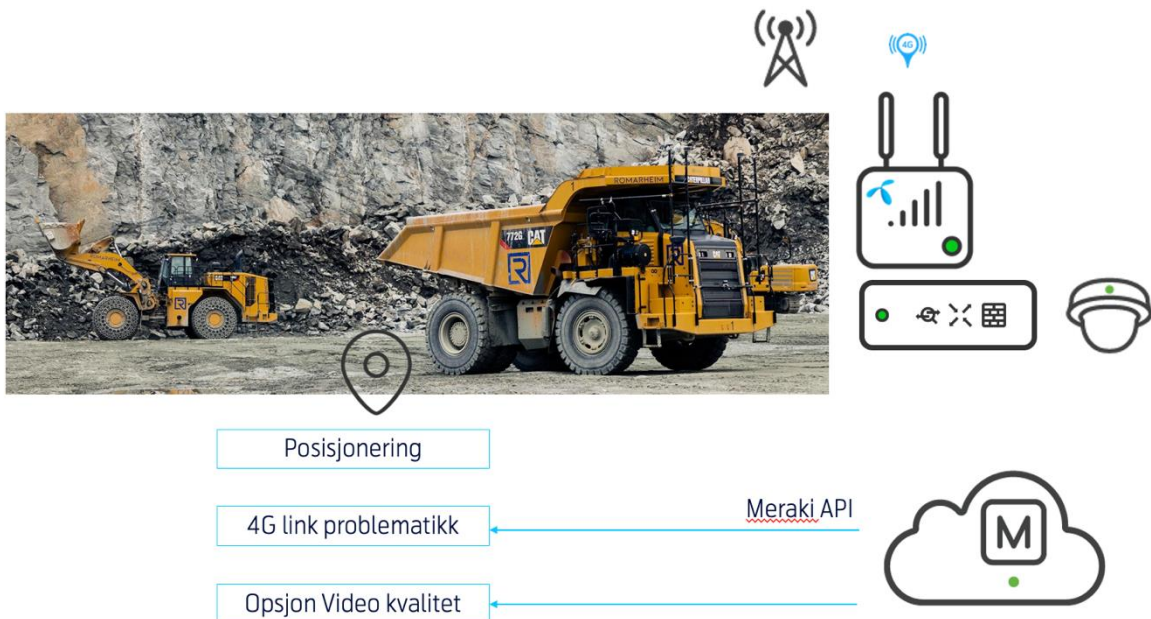
Problemstilling

I tiden fremover vil det være problemstillinger rundt hvordan sikre og avdekke «stans» i industrielle use-caser, som er sterkt knyttet til kvaliteten på den underliggende nettverksinfrastrukturen. Dette innebærer at kravet til robust og sikker nettverkskonnektivitet blir kritisk for industrien. For å levere en god nettverkstjeneste, er det avgjørende å sikre kvalitet gjennom monitorering og automatisering. På denne måten kan feilretting og tiltak iverksettes tidligere, for å sikre høy oppetid og god kvalitet på tjenesten.

Oppgave

Teste og lage en visuell kart oversikt over mobildekningen for et gitt geografisk område

- Benytte API fra Meraki, der man tar ut kvalitetsdata mtp. 4G mobildekning.
- Plotte dette inn i ett kart ala Google Maps eller lignende med riktig GPS koordinat.
- Visualisere på kartet hvor god dekningen er i området kjøretøyet/mobilradioen beveger seg.



Teknologi og Testsenario - 4G

Oppgaven skal se på viktige parameter for å understøtte industrielle use-case

1) Et mobilnett kan ha kapasitetsforskjeller ift. posisjon for et gitt kjøretøy

Virkemidler er tilgang til "dekningsband" som gir god arealdekning. I dag benyttes en miks av 800MHz, 1800MHz og 2100MHz for 4G i Norge. Her vil Telenor sammen med student kartlegge dette i et avgrenset testområdet på Gjøvik.

2) Kartlegge hvilke 4G datalink parametere om kan hentes ut vi API spørringer

SNR (signal-to-noise ratio) sier noe om hvor godt brukerstyr «hører» radiosignalet fra nærmeste basestasjon. Begrepet **SINR** (signal-to-interference-plus-noise ratio) benyttes i stedet for SNR for å ivareta interferens, dvs «radiohygiene» mellom radiocellene.

SINR er ett styreparameter i mobilnettet for å sette riktig modulasjonsform. De ulike modulasjonsformene gir ulike datarate. Høy SINR gir høy datarate(bit/s). Det oppnås alltid høyest datarate når brukerstyret er nær basestasjonen. Dataraten vil falle når brukerstyret beveger seg bort fra basestasjonen.

3)Drøfte en «Kritisk NEDETID-parameter» for en industriell use-case

Det er flere forhold som gir utfordringer for en industriell use-case. Med Telenor briller på, skal studenten drøfte en superparameter som gir en god indikasjon på «nedetid» for en gitt industriell use-case. Denne kan Telenor også bruke som trigger i en SLA tilstandsklokke som rapportere kvalitet for et gitt kjøretøy i et definert geografisk område som en gruve, industripark, skogteig etc.

Validering av verktøy - sikre og avdekke kvalitetsutfordringer presentert i en kartappliasjon

- Sette seg inn i Meraki sin nettverksplattform
- Drøfte mekanismer for å lese av posisjon for målingene, manuelt eller automatisert
- Sette seg inn Meraki API og validere omfang av 4G link parameter som er tilgjengelig på Meraki sin plattform
- Plotte resultatene i en kartapplikasjon mtp. posisjon og 4G kvalitetsparametre.

Forslag til tidsramme

Januar

- Forprosjekt & Studentene setter seg inn i tematikken.
- Kick-off med Telenor og eksterne ressurser.
- Lesing rundt teknologier.
- Utkast til introduksjon for selve rapporten
- Utstyr ankommer studiestedene. Start leking/testing med utstyret

Februar

- Strukturere testing/ scenarioer
- Start testing
- Statusmøter/oppfølging med Telenor

Mars

- Fortsatt testing
- Rapportskriving
- Statusmøter/oppfølging med Telenor

April

- Testing & skriving fortsetter
- Statusmøter/oppfølging med Telenor

Mai

- Finpuss av rapport

Hva kan Telenor tilby?

Veiledning fra fagmiljøet og dagers workshop

Fra Telenor sin side så har vi et ønske om å ha god dialog med studenten i hele prosessen. I januar organiserer Telenor et oppstartsmøte og diskuterer hvordan studenten ønsker å legge opp dette, og hva studenten ser for seg av behov for bistand og struktur. Det blir gjennomført en kick-off der man får en innføring i problematikken fra fagmiljøene Telenor og Cisco/Meraki. Det er etablert god kontakt med leverandørene her, og vil studentene vil ha kort vei til ekspertisen når oppgaven skrider frem.

Telenor og NTNU, gjennom Cisco Academy, arbeider målrettet med opplæring som treffer

behovet i markedet. Dette vil også lede til relevante sertifiseringer når studenten kommer ut i arbeidslivet.

Telenor sin anvendelse

Telenor vil benytte oppgaven til partnerutvikling mot Cisco.


Oppgaven skal føre til en presentasjon rundt temaet.

Det er ønskelig at det lages en playbook for ett demo oppsett, som kan benyttes for kunder på Telenor Expo. Telenor Expo er utstillingsvinduet til Telenor på Fornebu.

Campusnettverk- og internettarkitekturfordyping fra høstsemestret er en forutsetning for å ta denne oppgaven.


Telenor Team

Team




Bjørn Isene
Senior konsulent

✉ bjorn.isene@telenor.com
☎ +47 413 18 031
👤 0 reports



Andreas Rømo
Nettverkskonsulent

✉ andreas.romo@telenor.no
☎ +47 977 23 184
👤 0 reports



Henning Elvestad
Technical Sales Advisor

✉ henning.elvestad@telenor.com
☎ +47 412 45 932
👤 0 reports

Appendix B

Project Plan

This appendix contains the project plan for the project.



Kunnskap for en bedre verden

DEPARTMENT OF COMPUTER SCIENCE

DCSG2900 - BACHELOR THESIS BACHELOR OF SCIENCE IN
DIGITAL INFRASTRUCTURE AND CYBER SECURITY

Project Plan

Authors:

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

20th May 2023

Table of Contents

List of Figures	ii
List of Tables	ii
1 Introduction	1
1.1 Background	1
1.2 Assignment	1
2 Project Goals	1
2.1 Result Goals	1
2.2 Effect Goals	2
3 Scope	2
3.1 Time	2
3.2 Testing and Equipment	2
3.3 Product	2
3.4 Report	3
3.5 Theory	3
4 Project organization	3
4.1 Roles	3
4.2 Rules	3
4.3 Breach of contract	4
5 Planning, follow-up, reporting	4
5.1 Process framework	4
5.2 Plan for regular meetings	4
6 Organizing of quality assurance	5
6.1 Documentation, standards, communication channels and tools	5
6.1.1 Documentation	5
6.1.2 Standards	5
6.1.3 Communication channel	5
6.1.4 Tools	5
6.2 Plan for testing	6
6.2.1 Testing Area	6
6.2.2 Equipment	7

6.2.3	Software	8
6.2.4	Test Scenario	8
6.2.5	Test data	8
6.3	Risk analysis	9
6.3.1	Risk identification	9
6.3.2	Risk evaluation	10
6.3.3	Measures	10
7	Plan for implementation	11
7.1	Milestones	11
7.2	Gantt-chart	12
	Bibliography	13
	Appendix	14
A	Task description	15
B	Group Contract	19
C	Gantt-Chart	21

List of Figures

1	Map of the road used for testing	7
2	Map of the testing area with messurments from Telenor	7
3	Snippet from the gantt-chart.	12

List of Tables

1	Description of Consequences & Probability	9
2	Risk identification	9
3	Risk matrix levels explanation taken from (NTNU, 2013)	10
4	Risk matrix inspired by NTNUs regulation for risk analysis (NTNU, 2013)	10
5	Measures	11

1 Introduction

The bachelor thesis is the final evaluation we have as students in the 3 year bachelor's degree program Digital Infrastructure and Cyber Security at NTNU. We as students spend almost our entire final semester working on an assignment relevant to the program from an actual company and write a thesis about it. It is a good way to help us prepare for the actual work we will do after graduation. This document is the project plan for our bachelor thesis.

1.1 Background

The client for this thesis is Telenor, a company that among other things, maintain and expand the mobile network infrastructure in Norway. Telenor sees that it is critical to uncover quality challenges in the mobile network for industrial use-cases. One of these industrial use-cases where quality of the coverage of the mobile network is important is in self-driving tip-trucks working in quarries. The tip-trucks uses a combination of GPS from satellites and correction data received over the mobile network to get an accurate location of themselves. If they lose connection with the mobile network they can't get a accurate location and they stop and enter a safe mode. To improve this use-case, Telenor wants to find a good way to monitor the quality of the mobile network in these tip-trucks as the quarry gets deeper so they can get an early warning if the quality gets worse. This way they can improve the coverage before the tip-trucks lose connection, enters safe mode and stops.

1.2 Assignment

Ref. Task description in Appendix

Our assignment is to discuss and find a "super parameter" that can be used to give a good indication of "downtime" for an industrial use-case. This parameter should be based on the different parameters that affect the coverage and quality of the mobile network. The parameter should be able to be used to rapport on the quality of the network for a given vehicle in a given geographical area.

The second part of the assignment is to go out and take measurements and test the mobile network in a specific geographical area and develop a system that can take the results of these tests and put them into a map that show the quality of the area based on this "super parameter" that we found.

2 Project Goals

Here we will describe what goals we want to achieve with our bachelor project. We have divided the goals into effect goals and result goals. Result goal is specifically what we want to deliver at the end of project. Effect goals is what we want our project to achieve in the long-term, after the delivering the project.

2.1 Result Goals

Here are the result goals we as team has agreed upon:

- Produce a complete and well-written bachelor thesis.
- Produce a parameter which will indicate downtime and quality of network connectivity in specific industrial use-cases.

-
- Showcase all our mobile network coverage test results in a map-application, which will give a clear overview of the parameter readings we get from the meraki API.
 - Creating a well-documented playbook that can be displayed in Telenor Expo.

2.2 Effect Goals

Here are the effect goals we as team has agreed upon:

- Uncover problems for quality interference in network connectivity with mobile network.
- Telenor will be able to use the super parameter found through the project to gather data on network quality for remote controlled or self-driving vehicles in industrial use-cases.
- Contribute to partner development between Cisco and Telenor.
- Gain experience in conducting scientific assessment and being able to improve their reflective capabilities on all the findings.

3 Scope

To be able to do the assignment we need to establish a clear scope so that we don't stray from the objective.

3.1 Time

The time limit for when the entire project is to be finished is Monday May 22 2023. Beyond that we have as a group decided on some other deadlines for when parts of the project are to be finished, which can be seen in the Gant-chart.

3.2 Testing and Equipment

We will only test within the designated geographical area and only with the network equipment that we have been provided by the client, plus our own laptops. We will do the tests from a car driving through the area. The equipment will stay inside the car during the tests. We have decided to focus our testing on a section of Sknehovdvegen see Figure 1. We have chosen this section due to Telenor having data from previous tests that indicates varying signal quality on this road. For the tests themselves we will perform three tests on the same day, at different times. This is to see if the signal quality differs due to varying amount of data traffic. Weather conditions can also affect the signal quality, so if possible we will try to do some tests in bad weather. See section 6.2 for more on our testing procedures.

3.3 Product

The product we are going to deliver to the client will consist of a system that will showcase the test result visualized in a heatmap of the designated area and a written report.

There will also be the playbook containing the demo-setup which includes our detailed approach of how to replicate one of our testscenarios, including how to configure the hardware and setting up the heatmap in Elasticsearch using the results from the test.

3.4 Report

The report will include how and when we did the tests and why, and the following results and how we use those results to possibly tweak the next test to give some useful results that can help us get closer to figure out the super-parameter with analysis and discussion. It will also include a theoretical part.

3.5 Theory

The theoretical part will cover some basic theory around frequencies, signal, 4G LTE, measurement of 4G quality, radio propagation and other relevant theory related to the project so the reader of the report can get a better understanding of the main concepts talked about later in the report. The theory will be limited to what is required to cover the tests, the test results and the theory surrounding 4G that is required to better understand 4G.

4 Project organization

Ref. Group Contract in Appendix

4.1 Roles

We have decided to assign some permanent roles and responsibilities within the group. The roles are:

- Project manager - Espen-Andreas Oseth
Responsible for making sure that problems that arise within the group are being addressed and also that all group members have something useful to do.
- Referent - Lars Martin Nygaard
Responsible for writing a short summary of every meeting with the client and with the supervisor.
- Document manager - Vebjørn Albinson
Responsible for delivering the right document at the right time and having a local backup of the project.

In addition is Wisam Khalid Razi backup for every role, should the one having any of the roles become temporarily unavailable.

4.2 Rules

Some of the most essential rules that the group has agreed upon:

1. Decisions concerning the project are to be taken with a vote.
2. All expenses concerning the project that the client does not cover are to be split evenly between all group members.
3. Our primary work environment is to be the Git-lab page we made and documents are to be written using LaTeX.
4. Every group member are obliged to work at least 30 hours a week as of week 3, unless the group have decided something else for that specific week (though it is the average number of hours worked that is important, and that should average out to about 30 hours a week).

-
5. It is expected that each work hour logged consists of at least 50 minutes of work. This is mostly to make logging easier and make it easier to round off hours worked.
 6. Every group member is obliged to attend every meeting that the group have agreed upon, though exceptions can be made if there is good reason and the group has been alerted early (though there are a common understanding that in some cases early notice is not possible)
 7. There are to be written a short summary from each meeting with either client or supervisor.

4.3 Breach of contract

If any group member violates the rules agreed upon and signed by all members the routine is in escalating order:

1. Talks within the group to try to figure out a solution that the other group members can agree upon. If an agreement is reached, every group member is responsible for following up that the agreement is carried out.
2. Talks with the supervisor
3. Talks with the study manager and possible exclusion from the group. The client will be notified.

5 Planning, follow-up, reporting

To make steady progress in the project we need a plan for how we are going to do our tasks and what meetings we are going to have to ensure the continuous quality of the work and to react quickly if we stray from the assignment.

5.1 Process framework

We have concluded that the scientific method is the method that suits the task best. We will form a hypothesis on what we will observe, then we will do the experiment and analyze the resulting data to form new hypotheses that we can test by changing different parameters in the experiment and try it again to see if the results would be different. We can then make conclusions on how the different parameters affect the experiment. This way we will hopefully manage to find the super-parameter that the client is looking for to determine when a device is in danger of not getting good enough coverage.

We are also using a Kanban board on git to make the project workflow go as smoothly as possible. Kanban is an agile framework that aims to make the workflow more flexible and able to respond to change. A Kanban board works by having tasks move between categories as we work on them and makes it easier to manage the project (Kissflow, n.d.).

5.2 Plan for regular meetings

We have a regular status meeting every Monday at 09.00 to get an overview of what each group member has done and what status for the entire project is. It is also for quality control so that we can review the work that has been done to determine if the group agrees that it is good enough. Then we can determine what we are going to do the following week and divide the upcoming tasks between us.

We plan to have regular meetings with our advisor Thursday 10.00-10.30 every week. Meetings with the client will also be fairly regular, about every other week, but the weekday and time of day will vary depending on when the client has time.

6 Organizing of quality assurance

This section will describe the standards we have for documentation and which standards we will be using through the project. Risk analysis is also performed to identify the risks we may encounter in this project and identify the measures we can implement.

6.1 Documentation, standards, communication channels and tools

Here are the Documentation, standards, communication channels and tools we as a team will be using throughout this project.

6.1.1 Documentation

The document we produce throughout the project will be written in the Overleaf¹ platform, which allow the group to collaborate on the writing process. Overleaf requires writing all documents in LaTeX², which suits our needs to write long reports that contain figures and tables. Citations style we will be using is APA 7th³.

All the documents are kept on the overleaf platform, we will be using Gitlab-repository⁴ on NTNUs Git-lab to store a copy of the documents as well. Any source code and any other resources will also be stored in that Git-repository, where every team member has access. To secure further that we don't have any loss of data, we will have local backups of important documents and source code.

6.1.2 Standards

We will be using the issue board functionality in our GitLab-repository to create and hand out different task to team members. This will keep track of which tasks need to be done and gives an overview over tasks that has been completed. Every task needs to be reviewed by the team member to be marked as completed. These tasks can for example be writing a chapter in the rapport or setting up a test environment. This is very useful to keep track of what the team members are contributing and if the workload in the project is distributed equally.

Source code used should be of high quality, which means the code should be easy to read and understand, with clear and consistent variables and function names. The code should be robust and handle error, so it doesn't produce any faulty or unexpected result as well.

6.1.3 Communication channel

The main communication platform that will be used within the team, is the teams own Discord-server⁵. To communicate with our Telenor's contact people, we will be using a Microsoft Teams-channel⁶. Here every team member be able to see and communicate with the contacts. Document and other data will also be shared via the Teams-channel.

6.1.4 Tools

Here is a list over tools we will be using related to the project:

¹<https://www.overleaf.com/about#who-we-are>

²<https://www.latex-project.org/>

³<https://i.ntnu.no/oppgaveskriving/apa-7>

⁴<https://about.gitlab.com/>

⁵<https://discord.com/safety/360044149331-what-is-discord#title-1>

⁶<https://support.microsoft.com/en-us/office/first-things-to-know-about-channels-8e7b8f6f-0f0d-41c2-9883-3dc0bd5d4cda>

-
- Meraki Gateway MG21⁷: Cellular gateway used to connect to the mobile network and gather signal data
 - Meraki Dashboard API⁸: RESTful API that we will be using to get data from the Meraki gateway.
 - Nordic Thingy:91⁹: Multi-sensor cellular IoT prototyping platform we use to get GPS data and som signal data
 - Elasticsearch¹⁰ with Kibana¹¹: Application where we will insert the data gather from the Meraki gateway and the Nordic Thingy:91 into a heatmap
 - Docker¹²: Used to host containers with Elasticsearch and Kibana
 - SkyHigh¹³: NTNUs implementation of OpenStack¹⁴ where we run a Linux instance to run Docker.
 - LaTeX: Typesetting system we use to write the documentation for the project
 - OverLeaf: Online LaTeX editor that allows collaboration
 - GitLab¹⁵: GitLab is an implementation of Git used by NTNU and we use it to store files for the project and as a version control system
 - Microsoft Teams: Communication platform that we will use for communication with the client and our advisor.
 - Discord: Communication platform that we use as our main channel for communcation between team members.
 - Microsoft Visio¹⁶: Software we use to create diagrams and figures for the project
 - Python¹⁷: Programming language we use to create our tools for testing
 - Clockify¹⁸: Online Time tracking software.

6.2 Plan for testing

The objective of the testing is to measure the quality of the mobile network in a geographical area. We will then analyze the data and find a “super parameter” that can give us a good indication on the quality of the signals in the area. This data will then be put into Elasticsearch to be displayed in a map.

6.2.1 Testing Area

The tests will be performed in a specified geographical area found by Telenor. The area is a road where the quality of the mobile network is known to be a bit spotty in some places based on some tests from Telenor. Figure 1 shows a map of the testing area. During our first test we will analyze the data and determine the exact part of the area we will perform the tests based on the results.

⁷<https://meraki.cisco.com/product/cellular/integrated-antenna/mg21/>

⁸<https://developer.cisco.com/meraki/api-latest/#!/introduction/meraki-dashboard-api>

⁹<https://www.nordicsemi.com/Products/Development-hardware/Nordic-Thingy-91>

¹⁰<https://www.elastic.co/what-is/elasticsearch>

¹¹<https://www.elastic.co/kibana/>

¹²<https://www.docker.com/resources/what-container/>

¹³<https://www.ntnu.no/wiki/display/skyhigh>

¹⁴<https://www.openstack.org/software/>

¹⁵<https://about.gitlab.com/>

¹⁶<https://www.microsoft.com/en-us/microsoft-365/visio/flowchart-software>

¹⁷<https://www.python.org/about/>

¹⁸<https://clockify.me/>

We might have to use a larger area to get test results we can use or we might be able to do a smaller area.

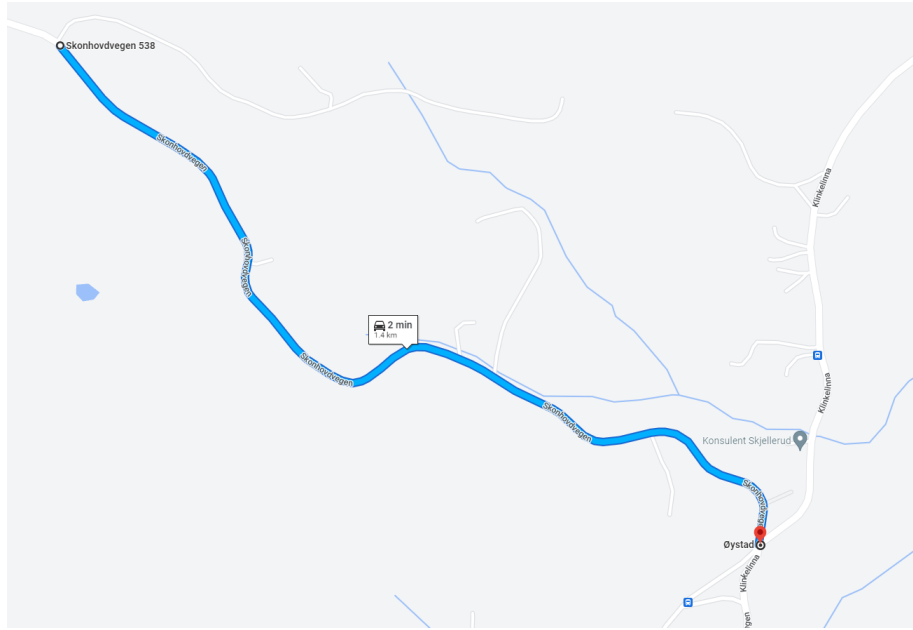


Figure 1: Map of the road used for testing

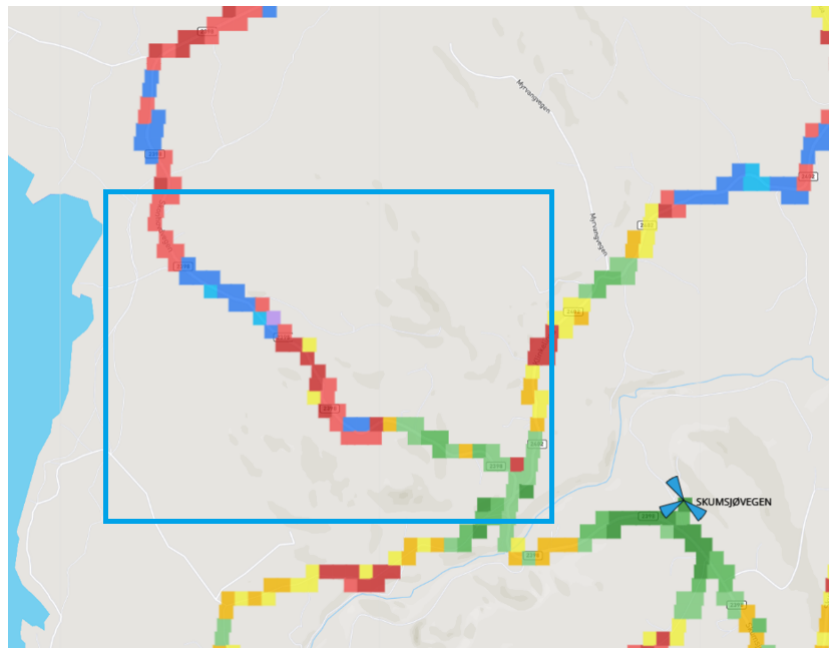


Figure 2: Map of the testing area with measurements from Telenor

6.2.2 Equipment

The equipment we will use for the test will be a Meraki Gateway MG21 and a Nordic Thingy:91. The Meraki Gateway MG21 is a device made by Meraki that connect to the mobile network and that connect to other device. The MG21 will send data about its connection and the quality of the signal it receives to the Meraki cloud service and show up in the Meraki Dashboard. We will then pull this data using the Meraki API with python. This will give us several different parameters, but the most important parameter for our testing is the connection type, RSRP and

RSRQ.

The Nordic Thingy:91 is a small device made by Nordic Semiconductors. This device will mainly be used for GPS and some signal parameters. We will pull data directly from this device so we will get the data in real time. This will be done by a simple C program.

We will also bring a laptop that the Meraki Gateway MG21 and the Nordic Thingy:91 connects to. This laptop will run the software for the test and collect the data from the devices as well as a Microsoft Teams videocall to another group member that is not in the car to get constant traffic. This will generate more data in the Meraki Dashboard we can use and give use a visual indication when the quality of the signal gets worse based on the video quality of the call.

6.2.3 Software

The software used for the testing will mainly be software we write ourselves. This will be a C program used to pull data from the Nordic Thingy:91 and a Python programmed used to get data from the Meraki API and then combine the data received from the Nordic Thingy:91 and the Meraki Gateway.

We will also use Microsoft Teams to generate data and give a visual indication of the quality of the connection and Elasticsearch with Kibana to organize and display the test data in a map.

6.2.4 Test Scenario

The tests will be done driving a car with the test equipment in the testing area. The Testing equipment will be mounted inside the car in the front of the vehicle. The equipment will be mounted in a way that they will be positioned the same way for every test and drive slowly at the same speed every time to get the most consistent results. During the drive the Meraki Gateway and the Nordic Thingy:91 will gather data and a Teams call will be running to generate traffic. We will pull data from the devices every 5 seconds during testing. This interval might be adjusted based on the results from the first tests, but should be consistent during all tests after we have adjusted it.

When performing the tests, we will take 3 tests on the same day. One test in the morning, one in the middle of the day or the afternoon and one test in the afternoon or during the night. This is to see if the quality of the mobile network will change during the day with different loads. We will also perform some tests during times where we believe we might get some different conditions like the start of the winter vacation and during bad weather if possible.

6.2.5 Test data

The data collected by the Meraki Gateway and the Nordic Thingy:91 will be collected on the laptop used during the testing before it gets combined, analyzed and displayed in a map. The biggest challenge with out setup is the Meraki Gateway uploads its data to a cloud service that we have to ask for the data. This means that there will be a delay between when the data gets recorded by the Meraki Gateway and when it is available in the Meraki Dashboard. The API will also just return the time when we asked for the data, not the time the data was recorded. We will have to do some testing to find a solution to this problem. The possible solutions we have is to either try to compare the data from the Meraki Gateway to the signal data we get in real time from the Nordic Thingy:91 and use this to try to line them up, or if the delay in the Meraki Dashboard is consistent, we might be able to just put a offset on the timestamp and get the data to line up this way.

After we have combined and lined up the data, it will be uploaded to Elasticsearch where we will store and analyze the data. Using Kibana with Elasticsearch we will also be able to put the test data into a map to visualize the mobile network quality in the testing area and compare it for different times.

6.3 Risk analysis

Risk analysis is a very useful to have when conducting any kind of project work. This is because there are many different obstacles that can jeopardize the success of the project or contribute to things not going as smoothly as they are supposed to. The risk analysis will give us a clear overview over the different risks we may face in the project period and how serious they might be.

6.3.1 Risk identification

Table 2 will showcase the possible risks we may encounter. Each risk will get probability score between 1-5, and also get a consequences score between 1-5. Table 1 will give detailed explanation of each level of probability and consequences.

Levels	Consequences	Probability
1	Minimal and won't affect the project progress	Highly improbable that the risk will occur
2	Impact the project progress to a small degree.	Improbable that the risk will occur
3	Severely impact the project and make it difficult to complete the project with the desired quality.	Probable risk we may encounter.
4	The impact is more severe and can lead to big problems in the project progress	Very probable that the risk will occur
5	Critical consequence which will make it very difficult to get the project approved	Highly improbable that the risk will occur

Table 1: Description of Consequences & Probability

Risk number	Description	Probability	Consequences
R1	Member(s) of the team gets affected by illness or have personal issues.	3	3
R2	Member(s) leave the team	1	4
R3	Test equipment breaks during tests	2	4
R4	Can't perform test scenarios in the designated test area due to roadwork or closed roads	2	3
R5	No access to car or driver to perform test scenarios	2	4
R6	The bachelor thesis is not delivered by deadline	1	5
R7	Losing access to data or online tool(s) used in the project. These tool(s) can be Git repository, Overleaf editor, Meraki API, skyhigh resources	2	4
R8	drift away from the project scope	3	3
R9	Bad communication between the team and the Telenor's contact person	2	4
R10	Breach of group contract	2	3

Table 2: Risk identification

6.3.2 Risk evaluation

We have also inserted each risk identified in table 2 in a risk matrix illustrated in Table 4. Each risk is placed in the risk matrix based on the value of the total risk and the R-number is corresponding with the number in table 2. Total risk is calculated by multiplying the probability value and consequence value of a risk and then placed based on total risk. Table 3 explains what the different colors in the risk matrix represent.

Green	Acceptable risk	Risk reducing measures can be assessed.
Yellow	Tolerable risk	Risk reducing measures must be assessed.
Red	Unacceptable risk	Risk reducing measures must be implemented

Table 3: Risk matrix levels explanation taken from (NTNU, 2013)

		Probability				
		Highly improbable (1)	Improbable (2)	Probable (3)	Very probable (4)	Highly probable (5)
Consequences	Critical (5)	R6				
	Severe (4)	R2	R3, R7, R9			
	Moderate (3)		R4, R5,R10	R1, R8		
	Minor (2)					
	Minimal (1)					

Table 4: Risk matrix inspired by NTNUs regulation for risk analysis (NTNU, 2013)

6.3.3 Measures

As we can see in Table 4 we can see that all the risks we have identified falls under tolerable risk in the risk matrix. Reference to Table 3, we must assess some risk reducing measures to reduce the probability or/and consequence of a risk, which will result in reducing the total risk for the risks

we may encounter. Table 5 will describe some of the measures we might implement if we find it necessary, as well as which of the risks are affected by these measures.

Measures	Risks affected
Divide the workload between the remaining team members, in case a team member(s) are unable to do their assigned work or a member(s) leaves the team	R1, R2
Locate another designated area where test scenarios can be performed in a similar environment as the original given test area	R4
Find a backup driver on standby to help perform the test scenarios, in case Espen-andreas oseth (only driver in the team) is not available due to illness or other reasons	R5
Locate a place where we can borrow a car in case of the car breaks down	R5
Meet all the deadlines found in the Gantt-chart	R6
Have local backups for documents, source code and important data related to the project	R7
Make the project scope as specific as possible and set clear boundaries on what to exclude and include in the project. This should be clarified early in the project plan.	R8
Organize frequent meetings between the team and Telenor's contact person. This is to contribute to better and more clear communication between the different parties	R9

Table 5: Measures

7 Plan for implementation

This section takes a look at our plan for implementation for the thesis. Included is a list of notable milestones and a snippet from our Gantt-chart. The plan is to follow the Gantt as closely as possible, however we are aware that there might be changes and additions made to the project underway.

7.1 Milestones

The following list contains our most notable milestones during the work on the thesis. The same milestones can be found in our Gantt-chart as well.

1. Project Plan Deadline **01.02.2023**
2. Finish first test **17.02.2023**

3. Delivery to advisor **05.04.2023**
4. Finish Playbook **21.04.2023**
5. Finish Testing **21.04.2023**
6. Report Deadline **22.05.2023**
7. Presentation **06-08.06.2023**

7.2 Gantt-chart

Ref. Gantt-chart in the appendix

This subsection contains a snippet of our gantt-chart, where you can see all the tasks, some time windows and a couple of milestones.

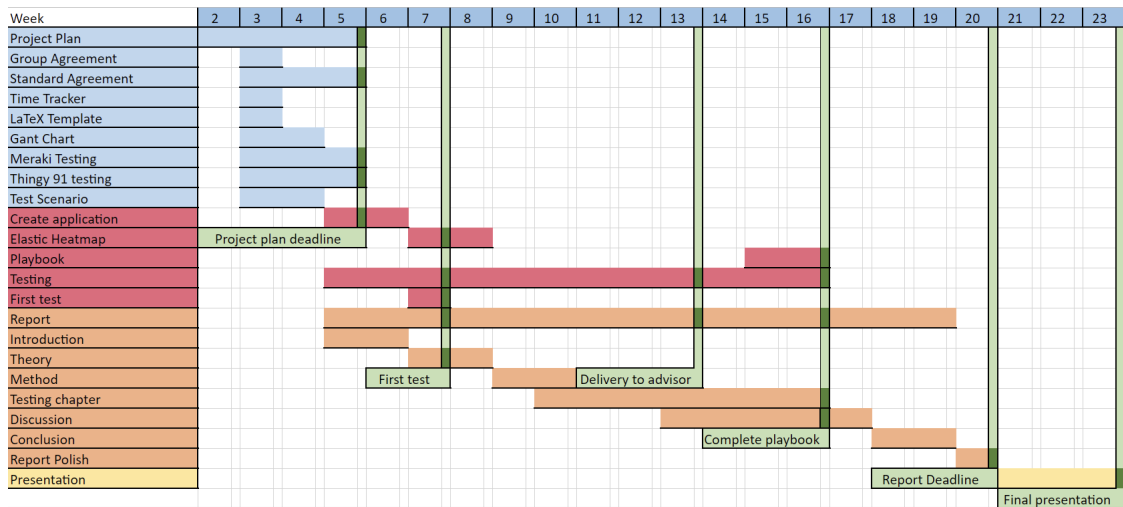


Figure 3: Snippet from the gantt-chart.

Bibliography

- Kissflow. (n.d.). *7 best project management methodologies and frameworks explained*. Retrieved 23rd January 2023, from <https://kissflow.com/project/project-management-methodologies-and-frameworks/>
- NTNU. (2013, October 31). *Gjennomføre risikovurdering*. Retrieved 19th January 2023, from <https://i.ntnu.no/wiki/-/wiki/Norsk/Gjennomf%C3%B8re+risikovurdering>

Appendix C

Scripts and Logs

All Scripts and logs used in this project can be found at: <https://github.com/bachelorTelenor/ScriptsAndLogs/tree/master>

The most important scripts for collection and processing data is included under:

C.1 Test Script

This script was used for collection data during testing. It gets data from the Meraki API and adds driving direction to the log.

Code listing C.1: Python script for combining logfiles

```
import os
import customtkinter
import json
import datetime
import csv
from datetime import datetime as dt
from itertools import islice

test = os.listdir("Logs")

customtkinter.set_appearance_mode("dark")
customtkinter.set_default_color_theme("dark-blue")

merakiLog = ""
iperfJson = ""
iperfLog = "./logs/iperfLog.csv"
gmonLog = ""
gpsLog = ""

combinedCSV = "./logs/CombinedLog.csv"

iperfStart = 1
gmonStart = 1
gpsStart = 1
```

```

def createCombineCsvFile():
    f = open(combinedCSV, 'w', newline='')
    writer = csv.writer(f, delimiter=';')
    csvHeader = ['MERAkIAPiquery', 'MERAkIAPiupdate', 'MERAkIrsrp', 'MERAkIrsrpPos',
                 'MERAkIrsrq', 'MERAkIrsrqPos', 'MERAkIband', 'TestDirection',
                 'IPERFtimestamp', 'IPERFmbs', 'GMONPLMN', 'GMONSYSTEM', 'GMONXCI',
                 'GMONxNBID', 'GMONLOCAL_CID', 'GMONLAC/TAC', 'GMONPCI/PSC/BSIC',
                 'GMONARFCN', 'GMONBAND', 'GMONBANDint', 'GMONRSSI', 'GMONRSRP',
                 'GMONRSrpPos', 'GMONRSRQ', 'GMONRSrqPos', 'GMONSNR', 'GMONCQI',
                 'GMONTA', 'GMONDISTANCE', 'GMONDELTA_AZI', 'GMONLAT', 'GMONLON',
                 'GMONLocation', 'GMONSPEED', 'GMONGPS_ACCURACY', 'GMONUL',
                 'GMONDL', 'GMONBANDWIDTH', 'GMONBANDWIDTHS', 'GMONCA',
                 'GMONNR_STATE', 'GMONNARFCN', 'GMONNR_BAND', 'GMONNR_PCI',
                 'GMONNR_SS_RSRP', 'GMONNR_SS_RSRQ', 'GMONNR_SS_SINR',
                 'GMONNR_CSI_RSRP', 'GMONNR_CSI_RSRQ', 'GMONNR_CSI_SINR',
                 'GMONCLF_LABEL', 'GMONCLF_LOC', 'GMONCLF_DESC', 'GMONDATE',
                 'GMONTIME', 'GMONTimestamp', 'GMONROAMING', 'GPStime', 'GPSlat',
                 'GPSlon', 'GPSLocation', 'GPSlevation', 'GPSaccuracy',
                 'GPSbearing', 'GPSspeed', 'GPSsatellites', 'GPSprovider',
                 'GPShdop', 'GPSvdop', 'GPSpdop', 'GPSgeoidheight',
                 'GPSageofdgpsdata', 'GPSdgpsid', 'GPSactivity', 'GPSbattery',
                 'GPSannotation', 'GPStimestamp_ms', 'GPStime_offset',
                 'GPSdistance', 'GPSstarttimestamp_ms', 'GPSprofile_name',
                 'GPSbattery_charging']
    writer.writerow(csvHeader)
    f.close()

def matchIperf(time):
    bestMatch = datetime.timedelta(hours=99999)
    match = False
    global iperfStart
    tempStart = iperfStart
    with open(iperfLog, 'r') as iperfData_csv:
        for l in islice(csv.reader(iperfData_csv), iperfStart, None):
            iperfTime = l[0]
            iperfdatedatetime = dt.strptime(iperfTime, '%Y-%m-%dT%H-%M-%SZ')
            iperfdatedatetime = iperfdatedatetime + datetime.timedelta(hours=1)
            timeDif = abs(time - iperfdatedatetime)

            if (timeDif < datetime.timedelta(seconds=5)):
                if (timeDif <= bestMatch):
                    bestMatch = timeDif

                    iperfTimestamp = \
                        iperfdatedatetime.strftime('%Y-%m-%dT%H:%M:%S+01')
                    iperfMBs = l[1]
            elif bestMatch != datetime.timedelta(hours=99999):
                match = True
                csvRow = [iperfTimestamp, iperfMBs]
                iperfStart = tempStart
                return csvRow, match
            elif time < (iperfdatedatetime + datetime.timedelta(seconds=5)):
                return 0, match
            if not time < (iperfdatedatetime - datetime.timedelta(seconds=2)):
                tempStart += 1
    return 0, match

def matchGmon(time):
    bestMatch = datetime.timedelta(hours=99999)

```

```

match = False
global gmonStart
tempStart = gmonStart
with open(gmonLog, 'r') as gmonData_csv:
    for l in islice(csv.reader(gmonData_csv, delimiter=';'), gmonStart, None):
        gmonTime = f'{l[39]}T{l[40]}Z'
        gmontimeDatetime = dt.strptime(gmonTime, '%Y/%m/%dT%H:%M:%SZ')
        timeDif = abs(time - gmontimeDatetime)

        if (timeDif < datetime.timedelta(seconds=5)):
            if (timeDif <= bestMatch):
                bestMatch = timeDif

                PLMN = l[0]
                SYSTEM = l[1]
                XCI = l[2]
                xNBID = l[3]
                LOCAL_CID = l[4]
                LACTAC = l[5]
                PCIPSCBSIC = l[6]
                ARFCN = l[7]
                BAND = l[8]
                BANDint = (BAND[0:4]).replace(' ', '')
                RSSI = l[9]
                RSRPRSCP = l[10]
                RsrpPos = abs(int(l[10]))
                RSRQECIO = l[11]
                RsrqPos = abs(int(l[11]))
                SNR = l[12]
                CQI = l[13]
                TA = l[14]
                DISTANCE = l[15]
                DELTA_AZI = l[16]
                LAT = l[17]
                LON = l[18]
                location = f'{LAT},{LON}'
                SPEED = l[19]
                GPS_ACCURACY = l[20]
                UL = l[21]
                DL = l[22]
                BANDWIDTH = l[23]
                BANDWIDTHS = l[24]
                CA = l[25]
                NR_STATE = l[26]
                NARFCN = l[27]
                NR_BAND = l[28]
                NR_PCI = l[29]
                NR_SS_RSRP = l[30]
                NR_SS_RSRQ = l[31]
                NR_SS_SINR = l[32]
                NR_CSI_RSRP = l[33]
                NR_CSI_RSRQ = l[34]
                NR_CSI_SINR = l[35]
                CLF_LABEL = l[36]
                CLF_LOC = l[37]
                CLF_DESC = l[38]
                DATE = l[39]
                TIME = l[40]
                timestamp = gmontimeDatetime.strftime('%Y-%m-%dT%H:%M:%S+01')
                ROAMING = l[41]

```

```

elif bestMatch != datetime.timedelta(hours=99999):
    match = True
    csvRow = [PLMN, SYSTEM, XCI, xNBID, LOCAL_CID, LACTAC, PCIPSCBSIC,
              ARFCN, BAND, BANDint, RSSI, RSRPRSCP, RsrpPos, RSRQECIO,
              RsrqPos, SNR, CQI, TA, DISTANCE, DELTA_AZI, LAT, LON,
              location, SPEED, GPS_ACCURACY, UL, DL, BANDWIDTH,
              BANDWIDTHS, CA, NR_STATE, NARFCN, NR_BAND, NR_PCI,
              NR_SS_RSRP, NR_SS_RSRQ, NR_SS_SINR, NR_CSI_RSRP,
              NR_CSI_RSRQ, NR_CSI_SINR, CLF_LABEL, CLF_LOC, CLF_DESC,
              DATE, TIME, timestamp, ROAMING]
    gmonStart = tempStart
    return csvRow, match
elif time < (gmontimeDatetime + datetime.timedelta(seconds=5)):
    return 0, match
if not time < (gmontimeDatetime - datetime.timedelta(seconds=2)):
    tempStart += 1
return 0, match

def matchGps(time):
    bestMatch = datetime.timedelta(hours=99999)
    global gpsStart
    tempStart = gpsStart
    match = False
    with open(gpsLog, 'r') as gpsData_csv:
        for l in islice(csv.reader(gpsData_csv), gpsStart, None):
            gpsTime = l[0]
            gpstimedatetime = dt.strptime(gpsTime, '%Y-%m-%dT%H:%M:%S.%fZ')
            gpstimedatetime = gpstimedatetime + datetime.timedelta(hours=2)
            timeDif = abs(time - gpstimedatetime)

            if (timeDif < datetime.timedelta(seconds=5)):
                if (timeDif <= bestMatch):
                    bestMatch = timeDif

                    time1 = gpstimedatetime.strftime('%Y-%m-%dT%H:%M:%S+01')
                    lat = l[1]
                    lon = l[2]
                    location = f'{{lat}},{{lon}}'
                    elevation = l[3]
                    accuracy = l[4]
                    bearing = l[5]
                    speed = l[6]
                    satellites = l[7]
                    provider = l[8]
                    hdop = l[9]
                    vdop = l[10]
                    pdop = l[11]
                    geoidheight = l[12]
                    ageofdgpsdata = l[13]
                    dgpsid = l[14]
                    activity = l[15]
                    battery = l[16]
                    annotation = l[17]
                    timestamp_ms = l[18]
                    time_offset = l[19]
                    distance = l[20]
                    starttimestamp_ms = l[21]
                    profile_name = l[22]
                    battery_charging = l[23]
            elif bestMatch != datetime.timedelta(hours=99999):

```

```

        match = True
        csvRow = [time1, lat, lon, location, elevation, accuracy, bearing,
                  speed, satellites, provider, hdop, vdop, pdop,
                  geoidheight, ageofdgpsdata, dgpsid, activity, battery,
                  annotation, timestamp_ms, time_offset, distance,
                  starttime_ms, profile_name, battery_charging]
        gpsStart = tempStart
        return csvRow, match
    elif time < (gpstimestamp + datetime.timedelta(seconds=5)):
        return 0, match
    if not time < (gpstimestamp - datetime.timedelta(seconds=2)):
        tempStart += 1
    return 0, match

def JsonToCsv():
    jsonfile = iperfJson
    csvpath = iperfLog

    f = open(csvpath, 'w', newline='')
    writer = csv.writer(f)
    csvHeader = ['Timestamp', 'Mb/s']
    writer.writerow(csvHeader)
    f.close()

    with open(jsonfile) as data_file:
        data = json.load(data_file)
        starttime = data['start']['timestamp']['timesecs']
        timestamp = datetime.datetime.fromtimestamp(starttime)
        timestamp = timestamp - datetime.timedelta(hours=1)
        orginalStart = timestamp
        for l in data["intervals"]:
            timestampPrint = timestamp.strftime("%Y-%m-%dT%H-%M-%S")
            Mbs = str(round((l['sum']['bits_per_second'] / 1024 / 1024), 2))
            csvRow = [timestampPrint, Mbs]
            f = open(csvpath, 'a', newline='')
            writer = csv.writer(f)
            writer.writerow(csvRow)
            f.close()
            timedelta = l['sum']['start']
            timestamp = orginalStart + datetime.timedelta(0, timedelta)

def combineLogs():
    JsonToCsv()
    createCombineCsvFile()
    with open(merakiLog, 'r') as merakiData_csv:
        merakiData = csv.reader(merakiData_csv)
        next(merakiData)
        for row in merakiData:
            merakiTime = row[1]
            merakiTime = dt.strptime(merakiTime, '%Y-%m-%dT%H:%M:%S')
            merakiTimeCSV = merakiTime.strftime('%Y-%m-%dT%H:%M:%S+01')
            merakiTime2 = row[0]
            merakiTime2 = dt.strptime(merakiTime2, '%Y-%m-%dT%H:%M:%S')
            merakiTimeCSV2 = merakiTime2.strftime('%Y-%m-%dT%H:%M:%S+01')
            merakiCSVrow = [merakiTimeCSV2, merakiTimeCSV, row[2], abs(int(row[2])),
                           row[3], abs(int(row[3])), row[4], row[5]]
            iperfCSVrow = matchIperf(merakiTime)
            gmonCSVrow = matchGmon(merakiTime)
            gpsCSVrow = matchGps(merakiTime)

```

```

    if iperfCSVrow[1] and gmonCSVrow[1] and gpsCSVrow[1]:
        merakiCSVrow.extend(iperfCSVrow[0])
        merakiCSVrow.extend(gmonCSVrow[0])
        merakiCSVrow.extend(gpsCSVrow[0])
        f = open(combinedCSV, 'a', newline='')
        writer = csv.writer(f, delimiter=';')
        writer.writerow(merakiCSVrow)
        f.close()

class MyTabView(customtkinter.CTkTabview):
    def __init__(self, master, **kwargs):
        super().__init__(master, **kwargs)

        # create tabs
        self.add("Meraki")
        self.add("Mobil")

        # ----- Meraki Tab -----
        # Meraki log select
        self.lableMeraki = customtkinter.CTkLabel(master=self.tab("Meraki"),
                                                  text="Meraki Log")
        self.lableMeraki.grid(row=0, column=0, padx=20, pady=20, sticky="ew")
        self.comboboxMeraki = customtkinter.CTkComboBox(master=self.tab("Meraki"),
                                                         values=test)
        self.comboboxMeraki.grid(row=0, column=1, columnspan=2, padx=20, pady=20,
                                  sticky="ew")

        # Iperf log select
        self.lableIperf = customtkinter.CTkLabel(master=self.tab("Meraki"),
                                                  text="Iperf Log (JSON)")
        self.lableIperf.grid(row=1, column=0, padx=20, pady=20, sticky="ew")
        self.comboboxIperf = customtkinter.CTkComboBox(master=self.tab("Meraki"),
                                                         values=test)
        self.comboboxIperf.grid(row=1, column=1, columnspan=2, padx=20, pady=20,
                                  sticky="ew")

        # GPSLogger log select
        self.lableGPSLogger = customtkinter.CTkLabel(master=self.tab("Meraki"),
                                                      text="GPSLogger Log")
        self.lableGPSLogger.grid(row=2, column=0, padx=20, pady=20, sticky="ew")
        self.comboboxGPSLogger = customtkinter.CTkComboBox(master=self.tab("Meraki"),
                                                            values=test)
        self.comboboxGPSLogger.grid(row=2, column=1, columnspan=2, padx=20, pady=20,
                                      sticky="ew")

        # Gmon log select
        self.lableGmon = customtkinter.CTkLabel(master=self.tab("Meraki"),
                                                 text="Gmon Log")
        self.lableGmon.grid(row=3, column=0, padx=20, pady=20, sticky="ew")
        self.comboboxGmon = customtkinter.CTkComboBox(master=self.tab("Meraki"),
                                                       values=test)
        self.comboboxGmon.grid(row=3, column=1, columnspan=2, padx=20, pady=20,
                                  sticky="ew")

        self.button = customtkinter.CTkButton(master=self.tab("Meraki"),
                                               command=self.button_callback,
                                               text="Combine")
        self.button.grid(row=4, column=0, columnspan=3, padx=20, pady=20,
                          sticky="ew")

```

```

# ----- Mobil Tab -----
# GPSLogger log select
self.lableGPSLogger2 = customtkinter.CTkLabel(master=self.tab("Mobil"),
                                              text="GPSLogger Log")
self.lableGPSLogger2.grid(row=0, column=0, padx=20, pady=20, sticky="ew")
self.comboboxGPSLogger2 = customtkinter.CTkComboBox(master=self.tab("Mobil"),
                                                    values=test)
self.comboboxGPSLogger2.grid(row=0, column=1, columnspan=2, padx=20, pady=20,
                              sticky="ew")

# Gmon log select
self.lableGmon2 = customtkinter.CTkLabel(master=self.tab("Mobil"),
                                          text="Gmon Log")
self.lableGmon2.grid(row=1, column=0, padx=20, pady=20, sticky="ew")
self.comboboxGmon2 = customtkinter.CTkComboBox(master=self.tab("Mobil"),
                                                values=test)
self.comboboxGmon2.grid(row=1, column=1, columnspan=2, padx=20, pady=20,
                        sticky="ew")

self.button2 = customtkinter.CTkButton(master=self.tab("Mobil"),
                                       command=self.button_callback2,
                                       text="Combine")
self.button2.grid(row=2, column=0, columnspan=3, padx=20, pady=20,
                  sticky="ew")

# ----- Funksjoner -----
def button_callback(self):
    global iperfJson
    global merakiLog
    global gmonLog
    global gpsLog
    iperfJson = f"./logs/{self.comboboxIperf.get()}"
    merakiLog = f"./logs/{self.comboboxMeraki.get()}"
    gmonLog = f"./logs/{self.comboboxGmon.get()}"
    gpsLog = f"./logs/{self.comboboxGPSLogger.get()}"
    combineLogs()

def button_callback2(self):
    test2 = f"GPS: {self.comboboxGPSLogger2.get()}\n" \
           f"Gmon: {self.comboboxGmon2.get()}"
    print(test2)

class App(customtkinter.CTk):
    def __init__(self):
        super().__init__()

        self.title("Combine and upload logs")

        self.tab_view = MyTabView(master=self)
        self.tab_view.grid(row=0, column=0, padx=20, pady=20)

if __name__ == "__main__":
    app = App()
    app.mainloop()

```

C.2 Combine Script

This script was used to combine and match the logs from testing to get a single complete log.

Code listing C.2: Python script used during testing

```
import meraki
import os
from dotenv import load_dotenv
from datetime import datetime
import customtkinter
import csv

customtkinter.set_appearance_mode("dark")
customtkinter.set_default_color_theme("dark-blue")

running = False
newrun = True
testlog = ""
updated = True
lastUpdateTime = ""
direction = "North"

# Loads variables from .env file
load_dotenv()
API_KEY = os.getenv('MERAKI_API_KEY')
# Variables for connection to Meraki API
dashboard = meraki.DashboardAPI(API_KEY, suppress_logging=True)
organization_id = os.getenv('MERAKI_ORG_ID')

# Gets data from the Meraki API
def getmerakidata():
    response = \
        dashboard.cellularGateway.getOrganizationCellularGatewayUplinkStatuses(
            organization_id, total_pages='all'
        )
    return response

# Used to check if the Meraki update time has updated
def checkupdate(time, newtime):
    if time == newtime:
        return False
    else:
        return True

# Creates the log file
def cretecsv():
    logTime = datetime.now().strftime("%Y-%m-%dT%H-%M-%SZ")
    testLog = f"TestLog{logTime}"
    csvHeader = ['APIQuery', 'APIUpdate', 'rsrp', 'rsrq', 'MerakiDirection']
    f = open(f"./{testLog}.csv", 'w', newline='')
    writer = csv.writer(f)
    writer.writerow(csvHeader)
    f.close()
    return testLog
```



```

class App(customtkinter.CTk):
    # Creates the GUI for the application
    def __init__(self):
        super().__init__()

        self.geometry("500x300")
        self.title("Meraki data")
        self.minsize(900, 400)

        # create 2x2 grid system
        self.grid_rowconfigure(0, weight=0)
        self.grid_rowconfigure(1, weight=1)
        self.grid_columnconfigure((0, 1), weight=1)

        self.label = customtkinter.CTkLabel(master=self,
                                             font=customtkinter.CTkFont(size=18),
                                             text="Stopped")
        self.label.grid(row=0, column=0, columnspan=2, padx=20, pady=(20, 0),
                        sticky="nsew")

        global direction

        self.labelDir = customtkinter.CTkLabel(master=self,
                                                font=customtkinter.CTkFont(size=18),
                                                text="North")
        self.labelDir.grid(row=1, column=0, padx=20, pady=20, sticky="nsew")

        self.button = customtkinter.CTkButton(master=self, text="Turn",
                                              command=self.turn,
                                              font=customtkinter.CTkFont(size=18))
        self.button.grid(row=1, column=1, padx=20, pady=20, sticky="nsew")

        self.textbox = customtkinter.CTkTextbox(master=self,
                                                font=customtkinter.CTkFont(size=18))
        self.textbox.grid(row=2, column=0, columnspan=2, padx=20, pady=(20, 0),
                          sticky="nsew")

        self.button = customtkinter.CTkButton(master=self, text="Start",
                                              command=self.start,
                                              font=customtkinter.CTkFont(size=18),
                                              fg_color="green")
        self.button.grid(row=3, column=1, padx=20, pady=20, sticky="ew")

        self.button = customtkinter.CTkButton(master=self, text="Stop",
                                              command=self.stop,
                                              font=customtkinter.CTkFont(size=18),
                                              fg_color="darkred")
        self.button.grid(row=3, column=0, padx=20, pady=20, sticky="ew")

        # Main function that gets the data from the API and write it to the log if
        # the API is updated
        # Runs every second
        def getdata(self):
            global running
            global newrun
            global testlog
            global updated
            global newUpdate
            global lastUpdateTime
            global direction

```

```

# Runs if start is pressed in the GUI
if running:
    # Runs the first loop after pressing start to create the log file
    # for the test
    if newrun:
        testlog = cretecsv()
        newrun = False
    # If the data from the API is updated, write it to the GUI and log file
    if updated:
        updated = False
        # Get the data from API
        merakiData = getmerakidata()
        lastUpdateTime = merakiData[0]["lastReportedAt"]
        currentTime = datetime.now().strftime("%Y-%m-%dT%H:%M:%SZ")
        rsrp = merakiData[0]["uplinks"][0]["signalStat"]["rsrp"]
        rsrq = merakiData[0]["uplinks"][0]["signalStat"]["rsrq"]

        # Writes to logfile
        csvRow = [currentTime, lastUpdateTime, rsrp, rsrq, direction]
        f = open(f"./{testlog}.csv", 'a', newline='')
        writer = csv.writer(f)
        writer.writerow(csvRow)
        f.close()

        newUpdate = f"API Query: {currentTime} | API update: " \
                    f"{lastUpdateTime} | rsrp: {rsrp} | rsrq: " \
                    f"{rsrq} | Direction: {direction}"
        # Writes the data to the GUI
        self.textbox.insert("0.0", f"{newUpdate}\n\n")
    # Loops to check if the API data is updated
    if not updated:
        app.update()
        merakiUpdate = getmerakidata()
        updateTime = merakiUpdate[0]["lastReportedAt"]
        updated = checkupdate(lastUpdateTime, updateTime)
    app.after(1000, self.getdata)

# Function for the start button. Starts the test
def start(self):
    global running
    running = True
    self.label.configure(text="Running")

# Function for the stop button. Stops the test
def stop(self):
    global running
    global newrun
    running = False
    newrun = True
    self.label.configure(text="Stopped")

# Function for the turn button. Change driving direction for the test
def turn(self):
    global direction
    if direction == "North":
        direction = "South"
        self.labelDir.configure(text="South")
    else:
        direction = "North"
        self.labelDir.configure(text="North")

```

```
if __name__ == "__main__":
    app = App()
    app.after(1000, app.getdata())
    app.mainloop()
```

C.3 Upload Script

This script was used to create indexes and upload the test data to Elasticsearch.

Code listing C.3: Python script used to create indexes and upload data to Elasticsearch

```
from requests.auth import HTTPBasicAuth
from elasticsearch import Elasticsearch, helpers
import csv
import os
from dotenv import load_dotenv

# Laster inn .env filen
load_dotenv()

ElasticSearchPass = os.getenv('ELASTIC_PASSWORD')
ElasticSearchIP = os.getenv('IP_ADDRESS')

client = Elasticsearch(
    ElasticSearchIP,
    ca_certs="./http_ca.crt",
    basic_auth=("elastic", ElasticSearchPass)
)

dataCSV = "Logs/KombinertLogTZ.csv"
dataCSVmobil = "Logs/KombinertLogMobilTZ.csv"

testscenario_index = {
    "properties": {
        "GMONARFCN": {
            "type": "long"
        },
        "GMONBAND": {
            "type": "keyword"
        },
        "GMONBANDWIDTH": {
            "type": "long"
        },
        "GMONBANDWIDTHS": {
            "type": "keyword"
        },
        "GMONBANDint": {
            "type": "long"
        },
        "GMONCA": {
            "type": "long"
        },
        "GMONCLF_DESC": {
```

```
"type": "text"
},
"GMONCLF_LABEL": {
  "type": "keyword"
},
"GMONCLF_LOC": {
  "type": "keyword"
},
"GMONCQI": {
  "type": "long"
},
"GMONDATE": {
  "type": "keyword"
},
"GMONDL": {
  "type": "long"
},
"GMONGPS_ACCURACY": {
  "type": "long"
},
"GMONLAC/TAC": {
  "type": "long"
},
"GMONLAT": {
  "type": "double"
},
"GMONLOCAL_CID": {
  "type": "long"
},
"GMONLON": {
  "type": "double"
},
"GMONLocation": {
  "type": "geo_point"
},
"GMONNR_STATE": {
  "type": "keyword"
},
"GMONPCI/PSC/BSIC": {
  "type": "long"
},
"GMONPLMN": {
  "type": "long"
},
"GMONROAMING": {
  "type": "keyword"
},
"GMONRSRP": {
  "type": "long"
},
"GMONRSRQ": {
  "type": "long"
},
"GMONRSSI": {
  "type": "long"
},
"GMONRsrpPos": {
  "type": "long"
},
"GMONRsrqPos": {
```

```
"type": "long"
},
"GMONSNR": {
  "type": "double"
},
"GMONSPEED": {
  "type": "long"
},
"GMONSYSTEM": {
  "type": "long"
},
"GMONTA": {
  "type": "long"
},
"GMONTIME": {
  "type": "keyword"
},
"GMONUL": {
  "type": "long"
},
"GMONXCI": {
  "type": "long"
},
"GMONtimestamp": {
  "type": "date",
  "format": "iso8601"
},
"GMONxNBID": {
  "type": "long"
},
"GPSLocation": {
  "type": "geo_point"
},
"GPSaccuracy": {
  "type": "double"
},
"GPSbattery": {
  "type": "long"
},
"GPSbattery_charging": {
  "type": "keyword"
},
"GPSbearing": {
  "type": "double"
},
"GPSdistance": {
  "type": "double"
},
"GPSelevation": {
  "type": "double"
},
"GPSgeoidheight": {
  "type": "long"
},
"GPShdop": {
  "type": "double"
},
"GPSlat": {
  "type": "double"
},
},
```

```
"GPSlon": {
  "type": "double"
},
"GPSpdop": {
  "type": "double"
},
"GPSprofile_name": {
  "type": "keyword"
},
"GPSprovider": {
  "type": "keyword"
},
"GPSsatellites": {
  "type": "long"
},
"GPSspeed": {
  "type": "double"
},
"GPSstarttimestamp_ms": {
  "type": "date",
  "format": "epoch_millis"
},
"GPStime": {
  "type": "date",
  "format": "iso8601"
},
"GPStime_offset": {
  "type": "date",
  "format": "iso8601"
},
"GPStimestamp_ms": {
  "type": "date",
  "format": "epoch_millis"
},
"GPSvdop": {
  "type": "double"
},
"IPERFmbs": {
  "type": "double"
},
"IPERFtimestamp": {
  "type": "date",
  "format": "iso8601"
},
"MERAKIAPIquery": {
  "type": "date",
  "format": "iso8601"
},
"MERAKIAPIupdate": {
  "type": "date",
  "format": "iso8601"
},
"MERAKIband": {
  "type": "long"
},
"MERAKIrsrp": {
  "type": "long"
},
"MERAKIrsrpPos": {
  "type": "long"
}
```

```
    },
    "MERAKIrsrq": {
      "type": "long"
    },
    "MERAKIrsrqPos": {
      "type": "long"
    },
    "TestDirection": {
      "type": "keyword"
    }
  }
}

client.indices.create(index='alltests', mappings=testscenario_idx)

testscenario_idx = {
  "properties": {
    "GMONARFCN": {
      "type": "long"
    },
    "GMONBAND": {
      "type": "keyword"
    },
    "GMONBANDWIDTH": {
      "type": "long"
    },
    "GMONBANDWIDTHS": {
      "type": "keyword"
    },
    "GMONBANDint": {
      "type": "long"
    },
    "GMONCA": {
      "type": "long"
    },
    "GMONCLF_DESC": {
      "type": "text"
    },
    "GMONCLF_LABEL": {
      "type": "keyword"
    },
    "GMONCLF_LOC": {
      "type": "keyword"
    },
    "GMONCQI": {
      "type": "long"
    },
    "GMONDATE": {
      "type": "keyword"
    },
    "GMONDL": {
      "type": "long"
    },
    "GMONGPS_ACCURACY": {
      "type": "long"
    },
    "GMONLAC/TAC": {
      "type": "long"
    },
    "GMONLAT": {
```

```
"type": "double"
},
"GMONLOCAL_CID": {
  "type": "long"
},
"GMONLON": {
  "type": "double"
},
"GMONLocation": {
  "type": "geo_point"
},
"GMONNR_STATE": {
  "type": "keyword"
},
"GMONPCI/PSC/BSIC": {
  "type": "long"
},
"GMONPLMN": {
  "type": "long"
},
"GMONROAMING": {
  "type": "keyword"
},
"GMONRSRP": {
  "type": "long"
},
"GMONRSRQ": {
  "type": "long"
},
"GMONRSSI": {
  "type": "long"
},
"GMONRsrpPos": {
  "type": "long"
},
"GMONRsrqPos": {
  "type": "long"
},
"GMONSNR": {
  "type": "double"
},
"GMONSPEED": {
  "type": "long"
},
"GMONSYSTEM": {
  "type": "long"
},
"GMONTA": {
  "type": "long"
},
"GMONTIME": {
  "type": "keyword"
},
"GMONUL": {
  "type": "long"
},
"GMONXCI": {
  "type": "long"
},
"GMONtimestamp": {
```



```
"type": "date",
"format": "iso8601"
},
"GMONxNBID": {
"type": "long"
},
"GPSLocation": {
"type": "geo_point"
},
"GPSaccuracy": {
"type": "double"
},
"GPSbattery": {
"type": "long"
},
"GPSbattery_charging": {
"type": "keyword"
},
"GPSbearing": {
"type": "double"
},
"GPSdistance": {
"type": "double"
},
"GPSelevation": {
"type": "double"
},
"GPSgeoidheight": {
"type": "long"
},
"GPShdop": {
"type": "double"
},
"GPSlat": {
"type": "double"
},
"GPSlon": {
"type": "double"
},
"GPSpdop": {
"type": "double"
},
"GPSprofile_name": {
"type": "keyword"
},
"GPSprovider": {
"type": "keyword"
},
"GPSsatellites": {
"type": "long"
},
"GPSspeed": {
"type": "double"
},
"GPSstarttimestamp_ms": {
"type": "date",
"format": "epoch_millis"
},
"GPStime": {
"type": "date",
```

```
        "format": "iso8601"
      },
      "GPStime_offset": {
        "type": "date",
        "format": "iso8601"
      },
      "GPStimestamp_ms": {
        "type": "date",
        "format": "epoch_millis"
      },
      "GPSvdop": {
        "type": "double"
      },
      "TestDirection": {
        "type": "keyword"
      }
    }
  }
}

client.indices.create(index='alltestsmobil', mappings=testscenario_index)

with open(dataCSV, 'r') as data_csv:
    reader = csv.DictReader(data_csv, delimiter=';')
    helpers.bulk(client, reader, index='alltests')

with open(dataCSVmobil, 'r') as data_csv2:
    reader2 = csv.DictReader(data_csv2, delimiter=';')
    helpers.bulk(client, reader2, index='alltestsmobil')
```


Appendix D

Demo Setup

Demo setup for a Elasticsearch and Kibana instance with test data from the project. All files for the Demo can be found at: <https://github.com/bachelorTelenor/DemoSetup>.

D.1 Setup Guide

A guide for setting up the demo.

bachelorTelenor / DemoSetup Public[Code](#) [Pull requests](#) [Actions](#) [Projects](#) [Security](#) [Insights](#) [Settings](#)DemoSetup / readme.md  vebjorn111 4 days ago

95 lines (73 loc) · 2.59 KB

Preview

Code

Blame



Elasticsearch demo setup

Prerequisites

Prerequisites for this demo

Python

<https://www.python.org/>

Check if installed by running:

```
python --version
```



Docker Compose

<https://docs.docker.com/compose/install/>

Check if installed by running:

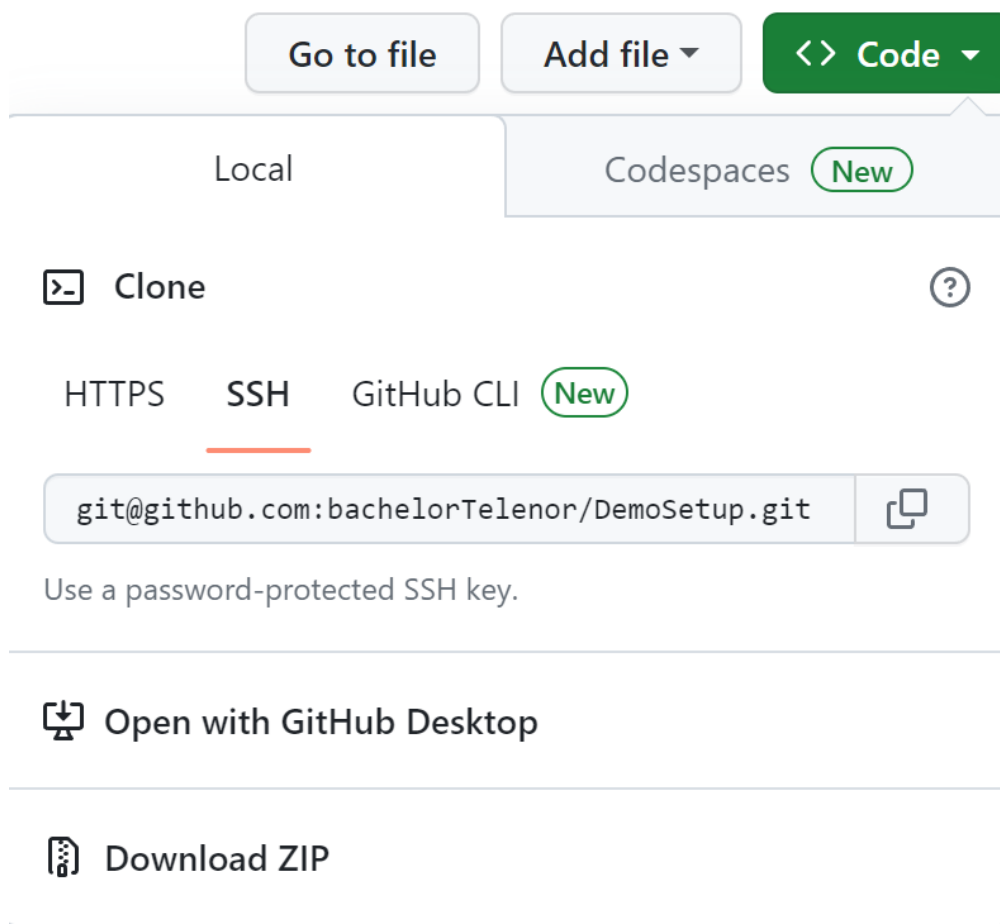
```
docker compose version
```



Installation

Get the installation files from Github

<https://github.com/bachelorTelenor/DemoSetup>



Go to file Add file <> Code

Local Codespaces New

Clone ?

HTTPS SSH GitHub CLI New

git@github.com:bachelorTelenor/DemoSetup.git

Use a password-protected SSH key.

Open with GitHub Desktop

Download ZIP

Go to the GitHub page and press the green button at the top of the page that says Code . Press Download ZIP to get a ZIP file container all the files. Extract the the ZIP file to get a directory called DemoSetup-master .

Open a command line interface like Terminal, CMD or PowerShell. Navigate to the DemoSetup-master directory. All commands in this guide will be run from this location.

Increase the virtual memory limit

Windows

```
wsl -d docker-desktop
sysctl -w vm.max_map_count=262144
exit
```



Linux

```
sysctl -w vm.max_map_count=262144
```



Start docker containers

Start the docker container by running:

```
docker-compose up -d
```



The container will start and begin setting up Elasticsearch and Kibana. This might take a few minutes. You can check if the containers is running with:

```
docker ps
```



This should show two containers running. They should be called `demosetup-kibana-1` and `demosetup-es01-1` .

Install Python libraries

Install the required libraries by running:

```
pip install -r requirements.txt
```



Check if Elasticsearch and Kibana is working

Make sure Elasticsearch and Kibana is done with the setup before continuing. Go to <http://localhost:5601/>. If you get the login screen, you can continue.

Get Elasticsearch certificate

Get the http certificate from the Elasticsearch container by running:

```
docker cp demosetup-master-es01-1:/usr/share/elasticsearch/config/certs/ca/ca.crt .
```



If the container is not called `demosetup-es01-1` run `docker ps` and use the name of your Elasticsearch container instead.

Run the setup script

```
python Setup.py
```



Log in to Kibana and navigate to the dashboard

Go to <http://localhost:5601/> and log in with:

User: `elastic`

Pass: demodemo

Navigate to the dashboard using the hamburger menu in the top left. Click `Dashboard` under `Analytics`. Click the dashboard called `All Tests`.

Stop docker containers

To stop the docker containers run:

```
docker compose down
```



If you want to remove the volumes and images used by the container, run this instead:

```
docker compose down -v --rmi all
```



D.2 Docker compose

Docker compose file for the docker container used in the demo.

Code listing D.1: Docker compose file for demo containers

```

version: "2.2"

services:
  setup:
    image: docker.elastic.co/elasticsearch/elasticsearch:${STACK_VERSION}
    volumes:
      - certs:/usr/share/elasticsearch/config/certs
    user: "0"
    command: >
      bash -c '
        if [ x${ELASTIC_PASSWORD} == x ]; then
          echo "Set the ELASTIC_PASSWORD environment variable in the .env file";
          exit 1;
        elif [ x${KIBANA_PASSWORD} == x ]; then
          echo "Set the KIBANA_PASSWORD environment variable in the .env file";
          exit 1;
        fi;
        if [ ! -f config/certs/ca.zip ]; then
          echo "Creating CA";
          bin/elasticsearch-certutil ca --silent --pem -out config/certs/ca.zip;
          unzip config/certs/ca.zip -d config/certs;
        fi;
        if [ ! -f config/certs/certs.zip ]; then
          echo "Creating certs";
          echo -ne \
            "instances:\n\
            " - name: es01\n\
            "   dns:\n\
            "     - es01\n\
            "     - localhost\n\
            "   ip:\n\
            "     - 127.0.0.1\n\
          > config/certs/instances.yml;
          bin/elasticsearch-certutil cert --silent --pem -out config/certs/certs.zip --in config/certs/instances.yml;
          unzip config/certs/certs.zip -d config/certs;
        fi;
        echo "Setting file permissions"
        chown -R root:root config/certs;
        find . -type d -exec chmod 750 {\} \;;
        find . -type f -exec chmod 640 {\} \;;
        echo "Waiting for Elasticsearch availability";
        until curl -s --cacert config/certs/ca/ca.crt https://es01:9200 | grep -q "missing authentication credentials"; do
          echo "Setting kibana_system password";
          until curl -s -X POST --cacert config/certs/ca/ca.crt -u "elastic:${ELASTIC_PASSWORD}" -H "Content-Type: application/json" -d '{"password": "kibana_system"}'; do
            echo "All done!";
          done;
        done;
      '

  healthcheck:
    test: ["CMD-SHELL", "[ -f config/certs/es01/es01.crt ]"]
    interval: 1s
    timeout: 5s
    retries: 120

  es01:
    depends_on:

```

```

    setup:
      condition: service_healthy
    image: docker.elastic.co/elasticsearch/elasticsearch:${STACK_VERSION}
    volumes:
      - certs:/usr/share/elasticsearch/config/certs
      - esdata01:/usr/share/elasticsearch/data
    ports:
      - ${ES_PORT}:9200
    environment:
      - node.name=es01
      - cluster.name=${CLUSTER_NAME}
      - cluster.initial_master_nodes=es01
      - ELASTIC_PASSWORD=${ELASTIC_PASSWORD}
      - bootstrap.memory_lock=true
      - xpack.security.enabled=true
      - xpack.security.http.ssl.enabled=true
      - xpack.security.http.ssl.key=certs/es01/es01.key
      - xpack.security.http.ssl.certificate=certs/es01/es01.crt
      - xpack.security.http.ssl.certificate_authorities=certs/ca/ca.crt
      - xpack.security.transport.ssl.enabled=true
      - xpack.security.transport.ssl.key=certs/es01/es01.key
      - xpack.security.transport.ssl.certificate=certs/es01/es01.crt
      - xpack.security.transport.ssl.certificate_authorities=certs/ca/ca.crt
      - xpack.security.transport.ssl.verification_mode=certificate
      - xpack.license.self_generated.type=${LICENSE}
    mem_limit: ${MEM_LIMIT}
    ulimits:
      memlock:
        soft: -1
        hard: -1
    healthcheck:
      test:
        [
          "CMD-SHELL",
          "curl -s --cacert config/certs/ca/ca.crt https://localhost:9200 | grep -q 'missing authentication c
        ]
      interval: 10s
      timeout: 10s
      retries: 120

    kibana:
      depends_on:
        es01:
          condition: service_healthy
      image: docker.elastic.co/kibana/kibana:${STACK_VERSION}
      volumes:
        - certs:/usr/share/kibana/config/certs
        - kibanadata:/usr/share/kibana/data
      ports:
        - ${KIBANA_PORT}:5601
      environment:
        - SERVERNAME=kibana
        - ELASTICSEARCH_HOSTS=https://es01:9200
        - ELASTICSEARCH_USERNAME=kibana_system
        - ELASTICSEARCH_PASSWORD=${KIBANA_PASSWORD}
        - ELASTICSEARCH_SSL_CERTIFICATEAUTHORITIES=config/certs/ca/ca.crt
      mem_limit: ${MEM_LIMIT}
      healthcheck:
        test:
          [

```

```

        "CMD-SHELL",
        "curl -s -I http://localhost:5601 | grep -q 'HTTP/1.1 302 Found'",
    ]
    interval: 10s
    timeout: 10s
    retries: 120

volumes:
  certs:
    driver: local
  esdata01:
    driver: local
  kibanadata:
    driver: local

```

D.3 Setup Script

Script for setting up the demo.

Code listing D.2: Python script for setting up the demo

```

import requests
from requests.auth import HTTPBasicAuth
from elasticsearch import Elasticsearch, helpers
import csv

ELASTIC_PASSWORD = "demodemo"
dataCSV = "KombinertLogg.csv"
dataCSVmobil = "KombinertLoggMobil.csv"

# Create the client instance
client = Elasticsearch(
    "https://localhost:9200",
    ca_certs="ca.crt",
    basic_auth=("elastic", ELASTIC_PASSWORD)
)

testscenario_idx = {
    "properties": {
        "GMONARFCN": {
            "type": "long"
        },
        "GMONBAND": {
            "type": "keyword"
        },
        "GMONBANDWIDTH": {
            "type": "long"
        },
        "GMONBANDWIDTHS": {
            "type": "keyword"
        },
        "GMONBANDint": {
            "type": "long"
        },
        "GMONCA": {
            "type": "long"
        }
    }
}

```

```
},
"GMONCLF_DESC": {
  "type": "text"
},
"GMONCLF_LABEL": {
  "type": "keyword"
},
"GMONCLF_LOC": {
  "type": "keyword"
},
"GMONCQI": {
  "type": "long"
},
"GMONDATE": {
  "type": "keyword"
},
"GMONDL": {
  "type": "long"
},
"GMONGPS_ACCURACY": {
  "type": "long"
},
"GMONLAC/TAC": {
  "type": "long"
},
"GMONLAT": {
  "type": "double"
},
"GMONLOCAL_CID": {
  "type": "long"
},
"GMONLON": {
  "type": "double"
},
"GMONLocation": {
  "type": "geo_point"
},
"GMONNR_STATE": {
  "type": "keyword"
},
"GMONPCI/PSC/BSIC": {
  "type": "long"
},
"GMONPLMN": {
  "type": "long"
},
"GMONROAMING": {
  "type": "keyword"
},
"GMONRSRP": {
  "type": "long"
},
"GMONRSRQ": {
  "type": "long"
},
"GMONRSSI": {
  "type": "long"
},
"GMONRsrpPos": {
  "type": "long"
}
```

```
    },
    "GMONRsrqPos": {
      "type": "long"
    },
    },
    "GMONSNR": {
      "type": "double"
    },
    },
    "GMONSPEED": {
      "type": "long"
    },
    },
    "GMONSYSTEM": {
      "type": "long"
    },
    },
    "GMONTA": {
      "type": "long"
    },
    },
    "GMONTIME": {
      "type": "keyword"
    },
    },
    "GMONUL": {
      "type": "long"
    },
    },
    "GMONXCI": {
      "type": "long"
    },
    },
    "GMONtimestamp": {
      "type": "date",
      "format": "iso8601"
    },
    },
    "GMONxNBID": {
      "type": "long"
    },
    },
    "GPSLocation": {
      "type": "geo_point"
    },
    },
    "GPSaccuracy": {
      "type": "double"
    },
    },
    "GPSbattery": {
      "type": "long"
    },
    },
    "GPSbattery_charging": {
      "type": "keyword"
    },
    },
    "GPSbearing": {
      "type": "double"
    },
    },
    "GPSdistance": {
      "type": "double"
    },
    },
    "GPSelevation": {
      "type": "double"
    },
    },
    "GPSgeoidheight": {
      "type": "long"
    },
    },
    "GPShdop": {
      "type": "double"
    },
    },
    "GPSlat": {
```

```
"type": "double"
},
"GPSlon": {
  "type": "double"
},
"GPSpdop": {
  "type": "double"
},
"GPSprofile_name": {
  "type": "keyword"
},
"GPSprovider": {
  "type": "keyword"
},
"GPSsatellites": {
  "type": "long"
},
"GPSspeed": {
  "type": "double"
},
"GPSstarttimestamp_ms": {
  "type": "date",
  "format": "epoch_millis"
},
"GPStime": {
  "type": "date",
  "format": "iso8601"
},
"GPStime_offset": {
  "type": "date",
  "format": "iso8601"
},
"GPStimestamp_ms": {
  "type": "date",
  "format": "epoch_millis"
},
"GPSvdop": {
  "type": "double"
},
"IPERFmbs": {
  "type": "double"
},
"IPERFtimestamp": {
  "type": "date",
  "format": "iso8601"
},
"MERAKEAPIquery": {
  "type": "date",
  "format": "iso8601"
},
"MERAKEAPIupdate": {
  "type": "date",
  "format": "iso8601"
},
"MERAKEIband": {
  "type": "long"
},
"MERAKEIrsrp": {
  "type": "long"
},
}
```

```
        "MERAKIrsrpPos": {
          "type": "long"
        },
        "MERAKIrsrq": {
          "type": "long"
        },
        "MERAKIrsrqPos": {
          "type": "long"
        },
        "TestDirection": {
          "type": "keyword"
        }
      }
    }
  }
}

client.indices.create(index='alltests', mappings=testscenario_idx)

testscenario_idx = {
  "properties": {
    "GMONARFCN": {
      "type": "long"
    },
    "GMONBAND": {
      "type": "keyword"
    },
    "GMONBANDWIDTH": {
      "type": "long"
    },
    "GMONBANDWIDTHS": {
      "type": "keyword"
    },
    "GMONBANDint": {
      "type": "long"
    },
    "GMONCA": {
      "type": "long"
    },
    "GMONCLF_DESC": {
      "type": "text"
    },
    "GMONCLF_LABEL": {
      "type": "keyword"
    },
    "GMONCLF_LOC": {
      "type": "keyword"
    },
    "GMONCQI": {
      "type": "long"
    },
    "GMONDATE": {
      "type": "keyword"
    },
    "GMONDL": {
      "type": "long"
    },
    "GMONGPS_ACCURACY": {
      "type": "long"
    },
    "GMONLAC/TAC": {
      "type": "long"
    }
  }
}
```

```
},
"GMONLAT": {
  "type": "double"
},
"GMONLOCAL_CID": {
  "type": "long"
},
"GMONLON": {
  "type": "double"
},
"GMONLocation": {
  "type": "geo_point"
},
"GMONNR_STATE": {
  "type": "keyword"
},
"GMONPCI/PSC/BSIC": {
  "type": "long"
},
"GMONPLMN": {
  "type": "long"
},
"GMONROAMING": {
  "type": "keyword"
},
"GMONRSRP": {
  "type": "long"
},
"GMONRSRQ": {
  "type": "long"
},
"GMONRSSI": {
  "type": "long"
},
"GMONRsrpPos": {
  "type": "long"
},
"GMONRsrqPos": {
  "type": "long"
},
"GMONSNR": {
  "type": "double"
},
"GMONSPEED": {
  "type": "long"
},
"GMONSYSTEM": {
  "type": "long"
},
"GMONTA": {
  "type": "long"
},
"GMONTIME": {
  "type": "keyword"
},
"GMONUL": {
  "type": "long"
},
"GMONXCI": {
  "type": "long"
}
```



```
    },
    "GMONtimestamp": {
      "type": "date",
      "format": "iso8601"
    },
    "GMONxNBID": {
      "type": "long"
    },
    "GPSLocation": {
      "type": "geo_point"
    },
    "GPSaccuracy": {
      "type": "double"
    },
    "GPSbattery": {
      "type": "long"
    },
    "GPSbattery_charging": {
      "type": "keyword"
    },
    "GPSbearing": {
      "type": "double"
    },
    "GPSdistance": {
      "type": "double"
    },
    "GPSelevation": {
      "type": "double"
    },
    "GPSgeoidheight": {
      "type": "long"
    },
    "GPShdop": {
      "type": "double"
    },
    "GPSlat": {
      "type": "double"
    },
    "GPSlon": {
      "type": "double"
    },
    "GPSpdop": {
      "type": "double"
    },
    "GPSprofile_name": {
      "type": "keyword"
    },
    "GPSprovider": {
      "type": "keyword"
    },
    "GPSsatellites": {
      "type": "long"
    },
    "GPSspeed": {
      "type": "double"
    },
    "GPSstarttimestamp_ms": {
      "type": "date",
      "format": "epoch_millis"
    },
  },
```

```
        "GPStime": {
            "type": "date",
            "format": "iso8601"
        },
        "GPStime_offset": {
            "type": "date",
            "format": "iso8601"
        },
        "GPStimestamp_ms": {
            "type": "date",
            "format": "epoch_millis"
        },
        "GPSvdop": {
            "type": "double"
        },
        "TestDirection": {
            "type": "keyword"
        }
    }
}

client.indices.create(index='alltestsmobil', mappings=testscenario_idx)

with open(dataCSV, 'r') as data_csv:
    reader = csv.DictReader(data_csv, delimiter=';')
    helpers.bulk(client, reader, index='alltests')

with open(dataCSVmobil, 'r') as data_csv2:
    reader2 = csv.DictReader(data_csv2, delimiter=';')
    helpers.bulk(client, reader2, index='alltestsmobil')

url = 'http://localhost:5601/api/saved_objects/_import?createNewCopies=true'
myheader = {"kbn-xsrf": "true"}
myfiles = {'file': open('export.ndjson', 'rb')}
basic = HTTPBasicAuth('elastic', 'demodemo')

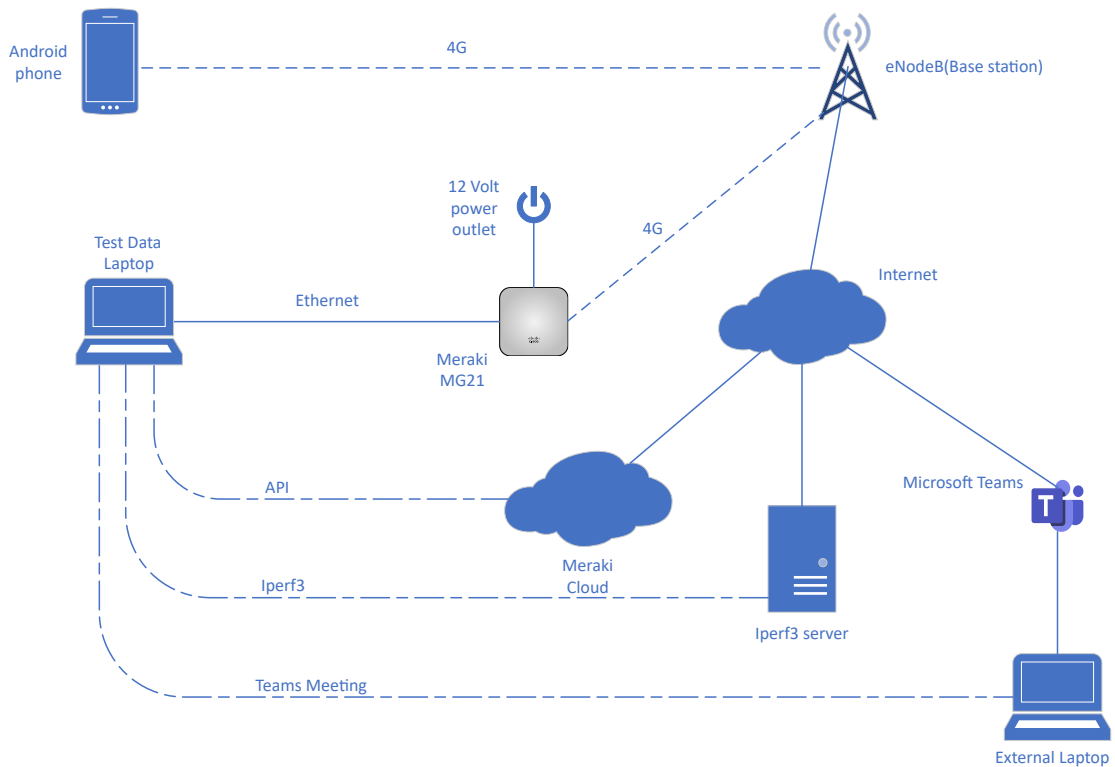
x = requests.post(url, headers=myheader, files=myfiles, auth=basic)
```

Appendix E

Documentation

This appendix contains simple documentation for the test setup and the Elastic-search setup used during the project. The files for the documentation are hosted on GitHub. <https://github.com/bachelorTelenor/Documentation>

Test Setup



Laptop

Python Script

The Python script `TestScript` was utilized to collect data from the Meraki API and record the driving direction of the test. To utilize the script, you need to fill out the `MERAKI_API_KEY` and `MERAKI_ORG_ID` in the `.env` file. The API key for `MERAKI_API_KEY` can be found under your profile on the Meraki Dashboard. You can obtain the organization ID for `MERAKI_ORG_ID` using the API and your API key. To utilize this functionality in Python, you can use:

```
import meraki

API_KEY = ''
# https://github.com/meraki/dashboard-api-python/

dashboard = meraki.DashboardAPI(API_KEY)

response = dashboard.organizations.getOrganizations()

print(response)
```

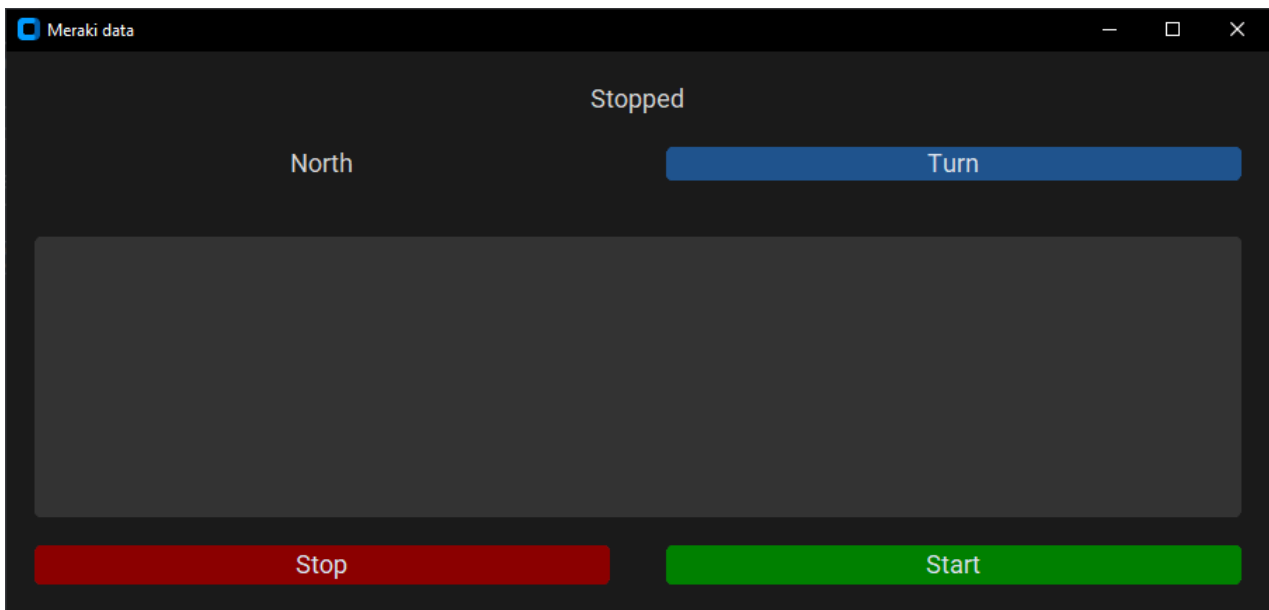


Install the required libraries for the script. The `requirements.txt` file contains the required libraries for the script. They can be installed by running the following command:

```
pip install -r requirements.txt
```



After the libraries are installed, you can run the script. You can start and stop logging with the Start and Stop buttons. When you stop the script and start it again, it will begin logging to a new file. Use the turn button to change the test direction between North and South. You can see if the script is logging or not at the top of the screen, and when new data is gathered, it will be displayed in the GUI.



iPerf3

<https://github.com/esnet/iperf>

iPerf3 was used during testing to log upload and download speeds of the MG21 connection. To use iPerf3, you need an iPerf3 server to test your connection. There are public iPerf3 servers available, but using a server you do not have control over can give inconsistent results.

To set up an iPerf3 server, download iPerf3 on the machine you want the server on and run:

```
./iperf3 -s
```



This will start an iPerf3 server on the default port 5201. You can find all the options for iPerf3 here if you want to change the default behavior: <https://iperf.fr/iperf-doc.php>.

Alternatively, you can run the iPerf3 server from a Docker container, as was done during this project. <https://hub.docker.com/r/networkstatic/iperf3>.

When the server is running, you can start the client. The option used during this project was:

```
.\iperf3.exe -c <SERVER_IP> -tinf -J --logfile testLog.json -P 4
```



This will initiate an upload test to `SERVER_IP` with 4 threads, logging the results in JSON format to the file `testLog.json`. The test will continue running until manually stopped. We have determined that utilizing multiple threads is necessary to achieve the maximum speed and fully utilize the connection's capacity. To test the download speed instead of upload, add the `-R` flag for reverse.

Web browser

We had a few browser windows open during testing to gather additional information. We had one window open with the Meraki dashboard displaying the frequency band it was using, and another window with a live clock to ensure accurate time for the test.

Microsoft Teams

Microsoft Teams was running during the tests with a video call to an external computer to maintain a continuous flow of data and to check if the user experience would be affected during testing.

OBS Studio

<https://obsproject.com/>

OBS was used on the laptop during testing to record the screen for analysis after the test.

Smartphone

GPSLogger

<https://github.com/mendhak/gpslogger/>

GPSLogger was used on the smartphone to log GPS data during testing. It was set to log to a CSV file with a logging interval of 1 second, a distance filter of 0 meters, and an accuracy filter of 10 meters.

G-MoN Pro

<https://play.google.com/store/apps/details?id=de.carknue.gmonpro&hl=en&gl=US>

G-MoN Pro was used on the smartphone to gather signal data during testing. This was done to obtain a secondary source of signal data for comparison with the MG21. G-MoN Pro also allowed for much shorter intervals between each data point, providing us with a greater amount of data. Additionally, G-MoN Pro was set to log data to a CSV file with an interval of 1 second.

Elasticsearch

For this project, Elasticsearch with Kibana was utilized to store and visualize the data from the tests. Elasticsearch was set up using Docker.

<https://www.elastic.co/guide/en/elasticsearch/reference/current/docker.html>

Prior to running the Docker container, it is necessary to increase the `vm.max_map_count` for Elasticsearch to function properly. This can be achieved by executing the following command:

```
sysctl -w vm.max_map_count=262144
```



There are two ways of setting up the Docker containers for Elasticsearch and Kibana. You can configure the containers separately and link Kibana to the Elasticsearch container, or you can use Docker-compose.

Docker Compose

Docker Compose is probably the easiest way to get Elasticsearch and Kibana running. The files needed for this can be found in the `docker-compose` directory. To get it up and running, simply modify the `.env` file to match your configuration and run the following command:

```
docker-compose up -d
```



This will run the container in the background and do the setup for you. When you want to stop the containers, use:

```
docker-compose down
```



Docker

To get Elasticsearch with Kibana up and running with just Docker is a bit more work than with Docker Compose.

Elasticsearch

Create a docker network for the containers, pull the image for Elasticsearch and run the container:

```
docker network create elastic
docker pull docker.elastic.co/elasticsearch/elasticsearch:8.6.0
docker run --name es-node01 --net elastic -p 9200:9200 -p 9300:9300 -t
docker.elastic.co/elasticsearch/elasticsearch:8.6.0
```



If you want to be able to reach Elasticsearch from another machine other than local host, set `network.publish_host`.

```
docker run --name es-node01 --net elastic -p 9200:9200 -p 9300:9300 -e network.publish_ho:
```



Kibana

Get the Kibana image and run the container:


```
docker pull docker.elastic.co/kibana/kibana:8.6.0
docker run --name kib-01 --net elastic -p 5601:5601 docker.elastic.co/kibana/kibana:8.6.0
```



On the first run, you will receive a link to the Kibana site, where you need to input the Elasticsearch enrollment token to connect Kibana to Elasticsearch. The next section will guide you on obtaining the enrollment token. In case you cannot access the provided link, you can access the Kibana container and add the enrollment token from there.

```
docker exec -it kib-01 bash
bin/kibana-setup --enrollment-token <your enrollment token here>
```



Generate enrollment tokens and passwords for Elasticsearch

Enrollment token:

```
docker exec -it es-node01 /usr/share/elasticsearch/bin/elasticsearch-create-enrollment-to
```



Password:

```
docker exec -it es-node01 /usr/share/elasticsearch/bin/elasticsearch-reset-password -u el
```



Connecting to Elasticsearch with Python

During the project, a certificate was used when connecting to Elasticsearch with Python. To obtain this certificate, run the following Docker command:

```
docker cp es-node01:/usr/share/elasticsearch/config/certs/http_ca.crt .
```



This command can look a bit different depending on how you set up the Docker containers. Both the container's name and the path to the certificate can change based on the method. In regular Docker, the path should be correct. If you used Docker Compose, the path should be `/usr/share/elasticsearch/config/certs/ca/ca.crt` instead. Once you have the certificate, you can connect to Elasticsearch with Python using it. Here's an example of how to test this:

```
from elasticsearch import Elasticsearch

ELASTIC_PASSWORD = "<password>"

client = Elasticsearch(
    "https://<HOST_IP>:9200",
    ca_certs="./http_ca.crt",
    basic_auth=("elastic", ELASTIC_PASSWORD)
)

print(client.info())
```



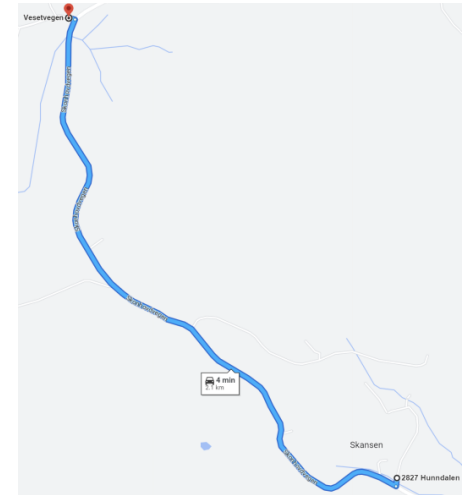
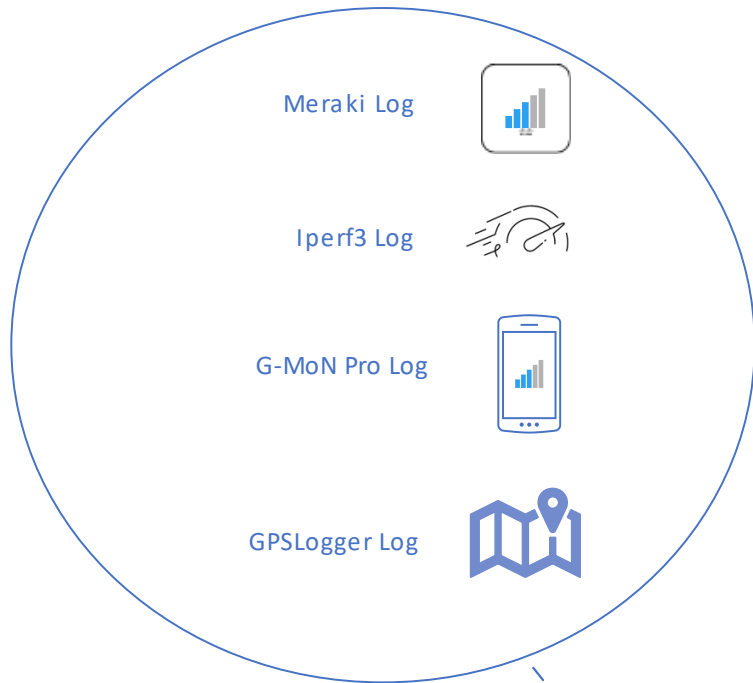
Appendix F

Playbook

Playbook for Elasticsearch. Demo setup for the playbook can be found in Appendix D.

F.1 Slides

Continuous measurement of the quality from A to B



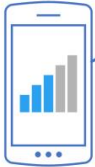
MG21 Log
(RSRP, RSRQ,
Frequency band)



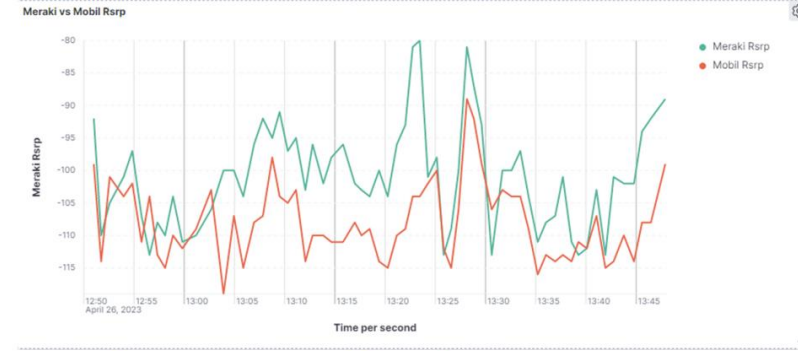
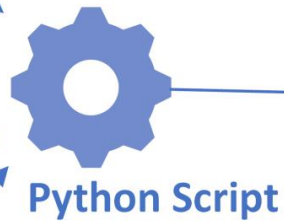
Iperf3 Log
(Download speed)



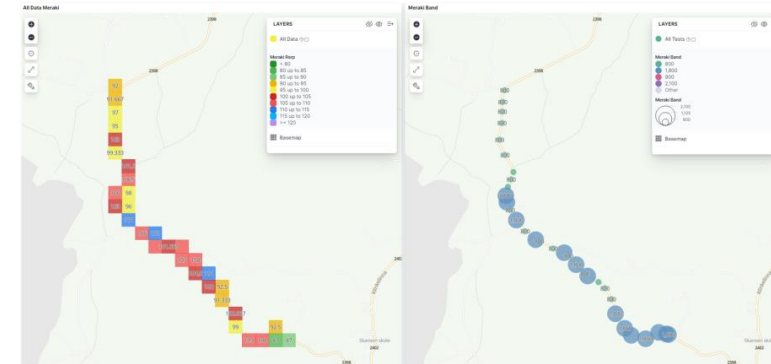
G-MoN Pro Log
(RSRP, RSRQ,
Frequency band)



GPSLogger Log
(GPS position)



Graphs



Heatmaps

F.2 Navigate Kibana

How to navigate in Kibana for demo purposes

1. Go to the Elastic home page, either through localhost or IP depending on your setup.
2. Log in to your user.
3. In the top left corner, click on the burger menu and select "Dashboard"
4. Select the dashboard you want to view.
5. Now you can see all the configured panels for the selected dashboard.
6. On the top section of the page you can access different filters.
 - The most important one, being the time filter.
 - Driving direction
 - Meraki and smartphone band
 - Meraki speed.
 - It is also possible to filter based on custom KQL commands.
7. To navigate in the maps click and hold left mouse click to pan the map, and hold control while scrolling to zoom in and out.
8. If you want to see a select panel in full screen. Click on the meatball menu in the top right corner of the window, select more and select maximize panel.
9. To exit the full screen follow the same steps above, but now maximize panel is swapped with minimize.
10. To edit the dashboard you need to click on the edit button on the top right of the dashboard.

Appendix G

Timesheet

Timesheet for each team member for the project

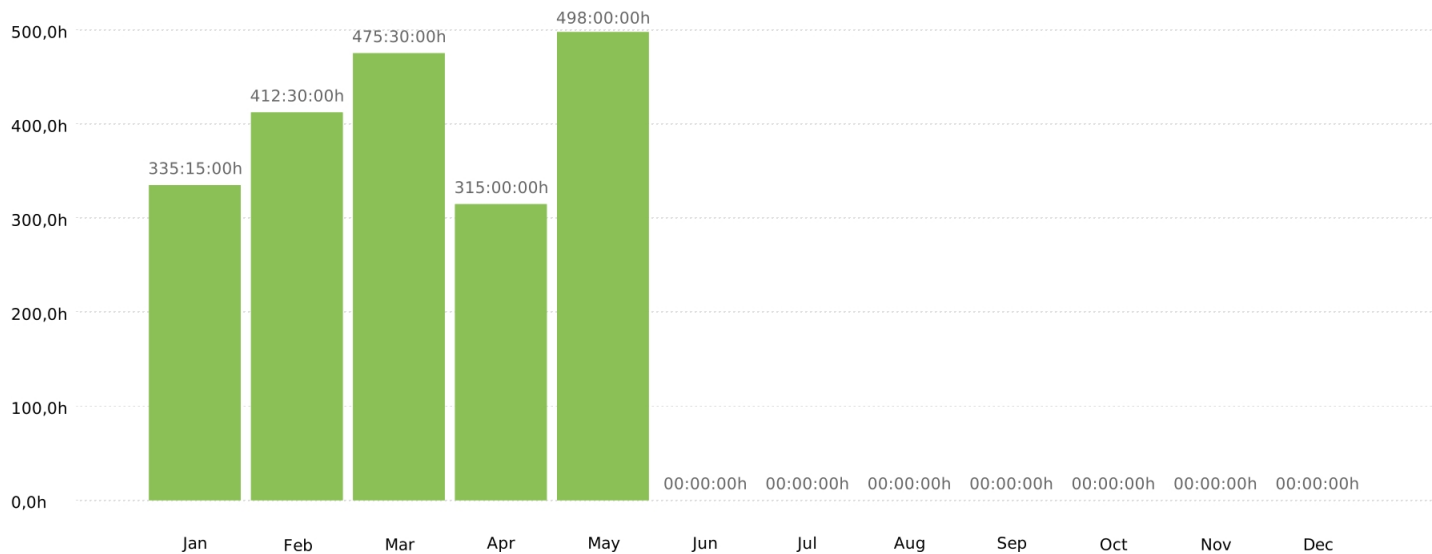
G.1 Overall

Summary report

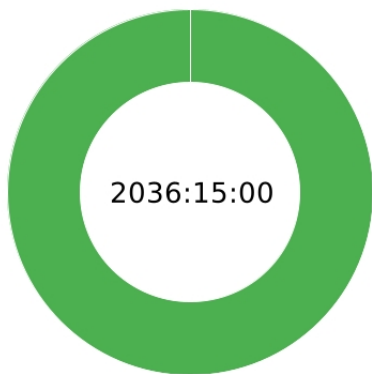
01/01/2023 - 31/12/2023



Total: 2036:15:00 Billable: 2036:15:00 Amount: 0,00 KR

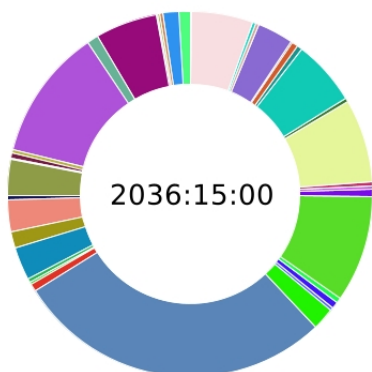


Project



● BachelorTelenor	2036:15:00	100,00%
-------------------	------------	---------

Tag



● (Without Tag)	19:30:00	0,96%
● ElasticSearch	29:30:00	1,45%
● ElasticSearch, Rapport, TestLogger	05:00:00	0,25%
● ElasticSearch, Research	04:00:00	0,20%
● ElasticSearch, TestLogger	03:00:00	0,15%

● Forprosjekt	110:45:00	5,44%
● Forprosjekt, Internt møte	21:00:00	1,03%
● Internt møte	236:30:00	11,61%
● Internt møte, ElasticSearch, TestOppsett	07:00:00	0,34%
● Internt møte, Forprosjekt	08:00:00	0,39%
● Internt møte, playbook	03:30:00	0,17%
● Internt møte, Rapport	64:30:00	3,17%
● Lesing bachelor/master oppgaver	07:30:00	0,37%
● Møte Telenor	57:30:00	2,82%
● Møte Veileder	29:30:00	1,45%
● playbook	58:00:00	2,85%
● playbook, Internt møte	08:00:00	0,39%
● playbook, Rapport	03:30:00	0,17%
● Prinsippskisse	14:30:00	0,71%
● Rapport	573:00:00	28,14%
● Rapport, Internt møte	39:30:00	1,94%
● Rapport, Research	06:30:00	0,32%
● Rapport, TestLogger	11:30:00	0,56%
● Rapport, TestOppsett	08:30:00	0,42%
● Research	193:30:00	9,50%
● Research, TestOppsett	12:00:00	0,59%
● Research, TestOppsett, testing	07:00:00	0,34%
● Research, Theory	07:00:00	0,34%
● testing	149:30:00	7,34%
● testing, TestOppsett	06:00:00	0,29%
● TestLogger	119:00:00	5,84%
● TestLogger, Internt møte	08:00:00	0,39%
● TestLogger, Rapport	12:00:00	0,59%

● TestLogger, TestOppsett	02:30:00	0,12%
● TestOppsett	65:30:00	3,22%
● TestOppsett, Internt møte	05:00:00	0,25%
● TestOppsett, playbook	06:00:00	0,29%
● Theory	111:00:00	5,45%
● Theory, Research	02:30:00	0,12%

Project / Tag	Duration	Amount
BachelorTelenor	2036:15:00	0,00 KR
(Without Tag)	19:30:00	0,00 KR
ElasticSearch	29:30:00	0,00 KR
ElasticSearch, Rapport, TestLogger	05:00:00	0,00 KR
ElasticSearch, Research	04:00:00	0,00 KR
ElasticSearch, TestLogger	03:00:00	0,00 KR
Forprosjekt	110:45:00	0,00 KR
Forprosjekt, Internt møte	21:00:00	0,00 KR
Internt møte	236:30:00	0,00 KR
Internt møte, ElasticSearch, TestOppsett	07:00:00	0,00 KR
Internt møte, Forprosjekt	08:00:00	0,00 KR
Internt møte, playbook	03:30:00	0,00 KR
Internt møte, Rapport	64:30:00	0,00 KR
Lesing bachelor/master oppgaver	07:30:00	0,00 KR
Møte Telenor	57:30:00	0,00 KR
Møte Veileder	29:30:00	0,00 KR
playbook	58:00:00	0,00 KR

playbook, Internt møte	08:00:00	0,00 KR
playbook, Rapport	03:30:00	0,00 KR
Prinsippskisse	14:30:00	0,00 KR
Rapport	573:00:00	0,00 KR
Rapport, Internt møte	39:30:00	0,00 KR
Rapport, Research	06:30:00	0,00 KR
Rapport, TestLogger	11:30:00	0,00 KR
Rapport, TestOppsett	08:30:00	0,00 KR
Research	193:30:00	0,00 KR
Research, TestOppsett	12:00:00	0,00 KR
Research, TestOppsett, testing	07:00:00	0,00 KR
Research, Theory	07:00:00	0,00 KR
testing	149:30:00	0,00 KR
testing, TestOppsett	06:00:00	0,00 KR
TestLogger	119:00:00	0,00 KR
TestLogger, Internt møte	08:00:00	0,00 KR
TestLogger, Rapport	12:00:00	0,00 KR
TestLogger, TestOppsett	02:30:00	0,00 KR
TestOppsett	65:30:00	0,00 KR
TestOppsett, Internt møte	05:00:00	0,00 KR
TestOppsett, playbook	06:00:00	0,00 KR
Theory	111:00:00	0,00 KR
Theory, Research	02:30:00	0,00 KR

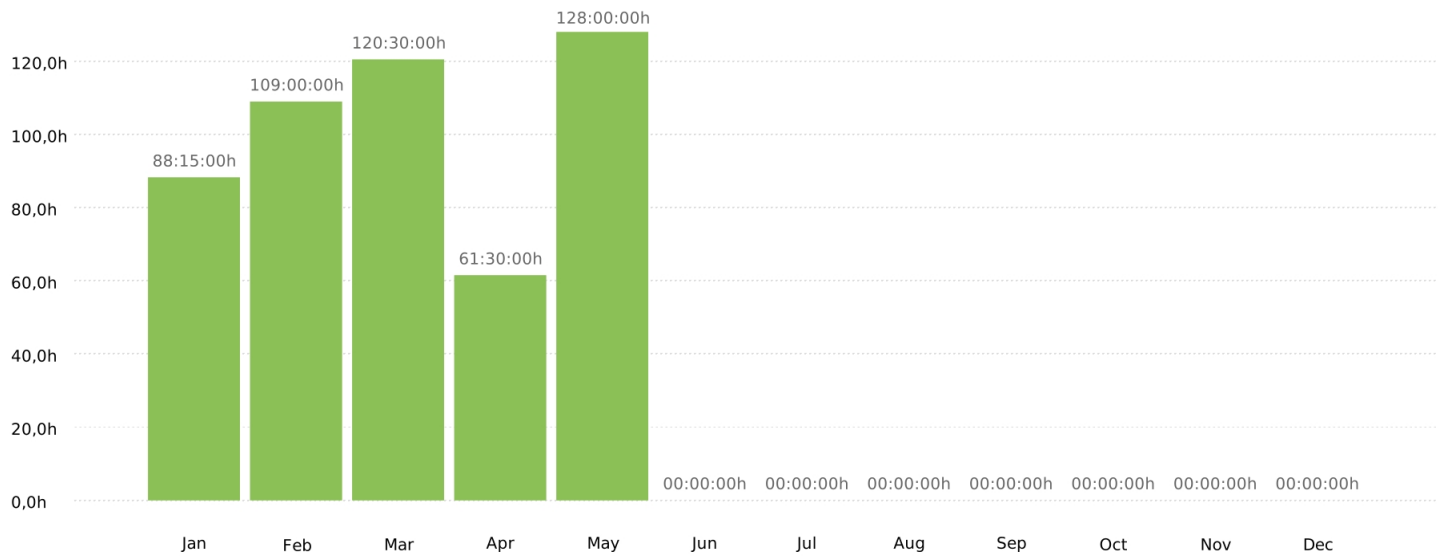
G.2 Espen-Andreas Oseth

Summary report

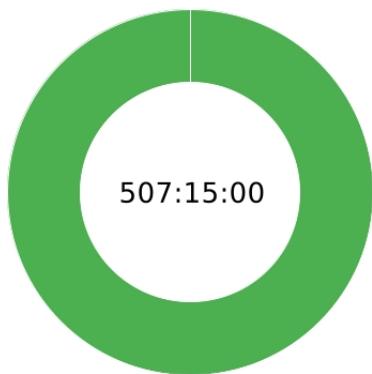
01/01/2023 - 31/12/2023



Total: 507:15:00 Billable: 507:15:00 Amount: 0,00 KR

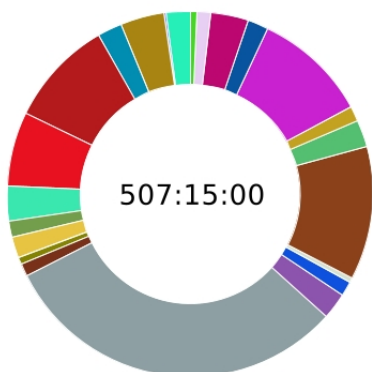


Project



● BachelorTelenor	507:15:00	100,00%
-------------------	-----------	---------

Tag



● (Without Tag)	10:30:00	2,07%
● ElasticSearch	01:00:00	0,20%
● Forprosjekt	19:45:00	3,89%
● Forprosjekt, Internt møte	11:00:00	2,17%
● Internt møte	47:30:00	9,36%

● Internt møte, Rapport	33:30:00	6,60%
● Møte Telenor	15:30:00	3,06%
● Møte Veileder	07:30:00	1,48%
● playbook	09:30:00	1,87%
● playbook, Internt møte	03:00:00	0,59%
● Prinsippskisse	05:30:00	1,08%
● Rapport	157:00:00	30,95%
● Rapport, Internt møte	12:00:00	2,37%
● Rapport, Research	06:30:00	1,28%
● Rapport, TestLogger	02:00:00	0,39%
● Research	60:00:00	11,83%
● Research, TestOppsett	12:00:00	2,37%
● Research, Theory	07:00:00	1,38%
● testing	51:00:00	10,05%
● TestLogger	09:30:00	1,87%
● TestOppsett	17:00:00	3,35%
● TestOppsett, testing	06:00:00	1,18%
● Theory	03:00:00	0,59%

Project / Tag	Duration	Amount
BachelorTelenor	507:15:00	0,00 KR
(Without Tag)	10:30:00	0,00 KR
ElasticSearch	01:00:00	0,00 KR
Forprosjekt	19:45:00	0,00 KR
Forprosjekt, Internt møte	11:00:00	0,00 KR
Internt møte	47:30:00	0,00 KR
Internt møte, Rapport	33:30:00	0,00 KR

Møte Telenor	15:30:00	0,00 KR
Møte Veileder	07:30:00	0,00 KR
playbook	09:30:00	0,00 KR
playbook, Internt møte	03:00:00	0,00 KR
Prinsippskisse	05:30:00	0,00 KR
Rapport	157:00:00	0,00 KR
Rapport, Internt møte	12:00:00	0,00 KR
Rapport, Research	06:30:00	0,00 KR
Rapport, TestLogger	02:00:00	0,00 KR
Research	60:00:00	0,00 KR
Research, TestOppsett	12:00:00	0,00 KR
Research, Theory	07:00:00	0,00 KR
testing	51:00:00	0,00 KR
TestLogger	09:30:00	0,00 KR
TestOppsett	17:00:00	0,00 KR
TestOppsett, testing	06:00:00	0,00 KR
Theory	03:00:00	0,00 KR

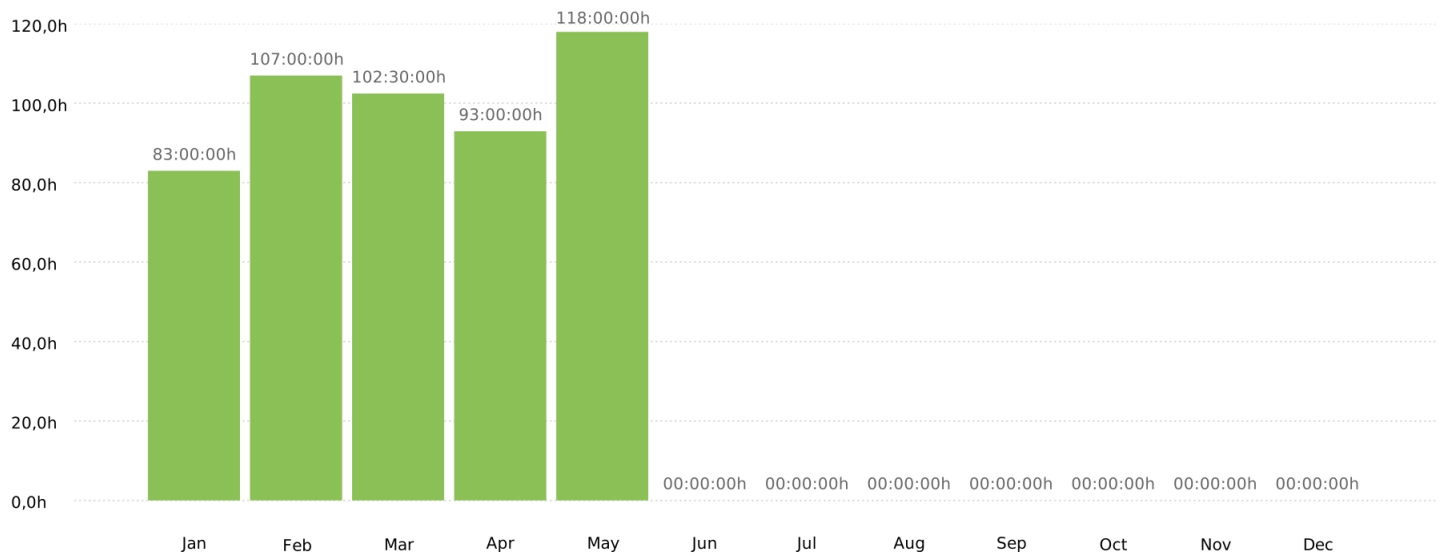
G.3 Lars Martin Nygaard

Summary report

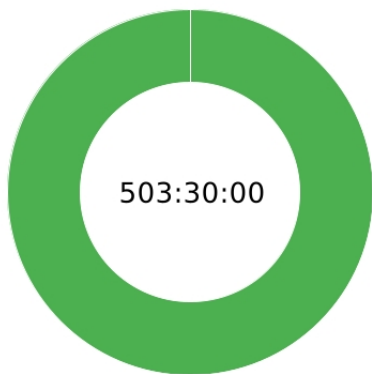
01/01/2023 - 31/12/2023



Total: 503:30:00 Billable: 503:30:00 Amount: 0,00 KR

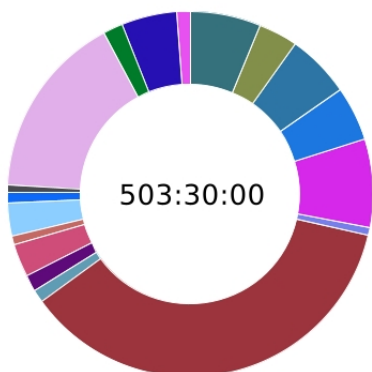


Project



● BachelorTelenor	503:30:00	100,00%
-------------------	-----------	---------

Tag



● (Without Tag)	06:00:00	1,19%
● Forprosjekt	24:30:00	4,87%
● Forprosjekt, Internt møte	09:00:00	1,79%
● Internt møte	82:30:00	16,39%
● Internt møte, Forprosjekt	03:00:00	0,60%

• Internt møte, playbook	05:00:00	0,99%
• Internt møte, Rapport	14:30:00	2,88%
• Lesing bachelor/master oppgaver	04:00:00	0,79%
• Møte Telenor	14:30:00	2,88%
• Møte Veileder	07:30:00	1,49%
• playbook	05:30:00	1,09%
• Rapport	183:30:00	36,44%
• Rapport, Internt møte	03:30:00	0,70%
• Research	39:00:00	7,75%
• testing	24:30:00	4,87%
• TestLogger	28:00:00	5,56%
• TestOppsett	17:30:00	3,48%
• Theory	31:30:00	6,26%

Project / Tag	Duration	Amount
BachelorTelenor	503:30:00	0,00 KR
(Without Tag)	06:00:00	0,00 KR
Forprosjekt	24:30:00	0,00 KR
Forprosjekt, Internt møte	09:00:00	0,00 KR
Internt møte	82:30:00	0,00 KR
Internt møte, Forprosjekt	03:00:00	0,00 KR
Internt møte, playbook	05:00:00	0,00 KR
Internt møte, Rapport	14:30:00	0,00 KR
Lesing bachelor/master oppgaver	04:00:00	0,00 KR
Møte Telenor	14:30:00	0,00 KR
Møte Veileder	07:30:00	0,00 KR

playbook	05:30:00	0,00 KR
Rapport	183:30:00	0,00 KR
Rapport, Internt møte	03:30:00	0,00 KR
Research	39:00:00	0,00 KR
testing	24:30:00	0,00 KR
TestLogger	28:00:00	0,00 KR
TestOppsett	17:30:00	0,00 KR
Theory	31:30:00	0,00 KR

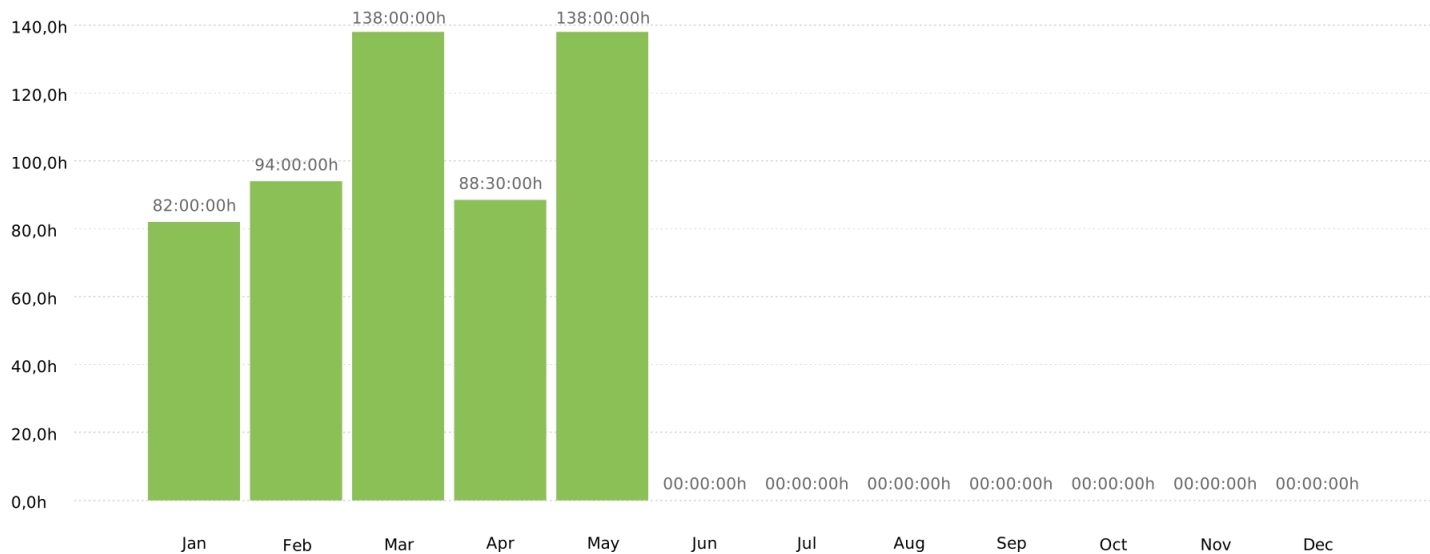
G.4 Vebjørn Albinson

Summary report

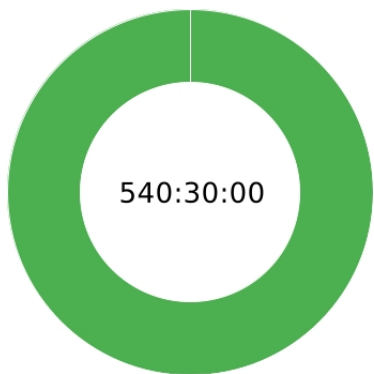
01/01/2023 - 31/12/2023



Total: 540:30:00 Billable: 540:30:00 Amount: 0,00 KR

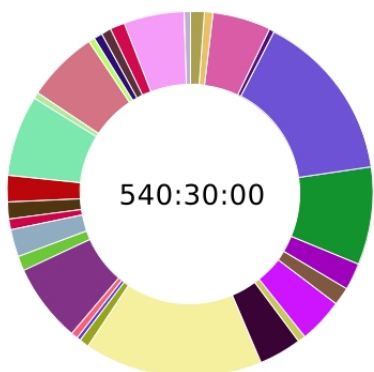


Project



● BachelorTelenor	540:30:00	100,00%
-------------------	-----------	---------

Tag



● (Without Tag)	03:00:00	0,56%
● ElasticSearch	28:30:00	5,27%
● ElasticSearch, Internt møte, TestOppsett	07:00:00	1,30%
● ElasticSearch, Rapport, TestLogger	05:00:00	0,93%
● ElasticSearch, Research	04:00:00	0,74%

● ElasticSearch, TestLogger	03:00:00	0,56%
● Forprosjekt	34:00:00	6,29%
● Forprosjekt, Internt møte	03:00:00	0,56%
● Internt møte	39:00:00	7,22%
● Internt møte, Rapport	12:30:00	2,31%
● Internt møte, TestLogger	08:00:00	1,48%
● Internt møte, TestOppsett	05:00:00	0,93%
● Møte Telenor	13:00:00	2,41%
● Møte Veileder	07:00:00	1,30%
● playbook	37:30:00	6,94%
● playbook, Internt møte	03:30:00	0,65%
● playbook, TestOppsett	02:00:00	0,37%
● Prinsippskisse	04:30:00	0,83%
● Rapport	85:00:00	15,73%
● Rapport, Internt møte	20:30:00	3,79%
● Rapport, playbook	03:30:00	0,65%
● Rapport, TestLogger	21:30:00	3,98%
● Rapport, TestOppsett	08:30:00	1,57%
● Research	13:00:00	2,41%
● testing	46:30:00	8,60%
● TestLogger	81:30:00	15,08%
● TestLogger, TestOppsett	02:30:00	0,46%
● TestOppsett	27:30:00	5,09%
● TestOppsett, playbook	04:00:00	0,74%
● TestOppsett, testing, Research	07:00:00	1,30%

Project / Tag	Duration	Amount
BachelorTelenor	540:30:00	0,00 KR
(Without Tag)	03:00:00	0,00 KR
ElasticSearch	28:30:00	0,00 KR
ElasticSearch, Internt møte, TestOppsett	07:00:00	0,00 KR
ElasticSearch, Rapport, TestLogger	05:00:00	0,00 KR
ElasticSearch, Research	04:00:00	0,00 KR
ElasticSearch, TestLogger	03:00:00	0,00 KR
Forprosjekt	34:00:00	0,00 KR
Forprosjekt, Internt møte	03:00:00	0,00 KR
Internt møte	39:00:00	0,00 KR
Internt møte, Rapport	12:30:00	0,00 KR
Internt møte, TestLogger	08:00:00	0,00 KR
Internt møte, TestOppsett	05:00:00	0,00 KR
Møte Telenor	13:00:00	0,00 KR
Møte Veileder	07:00:00	0,00 KR
playbook	37:30:00	0,00 KR
playbook, Internt møte	03:30:00	0,00 KR
playbook, TestOppsett	02:00:00	0,00 KR
Prinsippskisse	04:30:00	0,00 KR
Rapport	85:00:00	0,00 KR
Rapport, Internt møte	20:30:00	0,00 KR
Rapport, playbook	03:30:00	0,00 KR
Rapport, TestLogger	21:30:00	0,00 KR
Rapport, TestOppsett	08:30:00	0,00 KR

Research	13:00:00	0,00 KR
testing	46:30:00	0,00 KR
TestLogger	81:30:00	0,00 KR
TestLogger, TestOppsett	02:30:00	0,00 KR
TestOppsett	27:30:00	0,00 KR
TestOppsett, playbook	04:00:00	0,00 KR
TestOppsett, testing, Research	07:00:00	0,00 KR

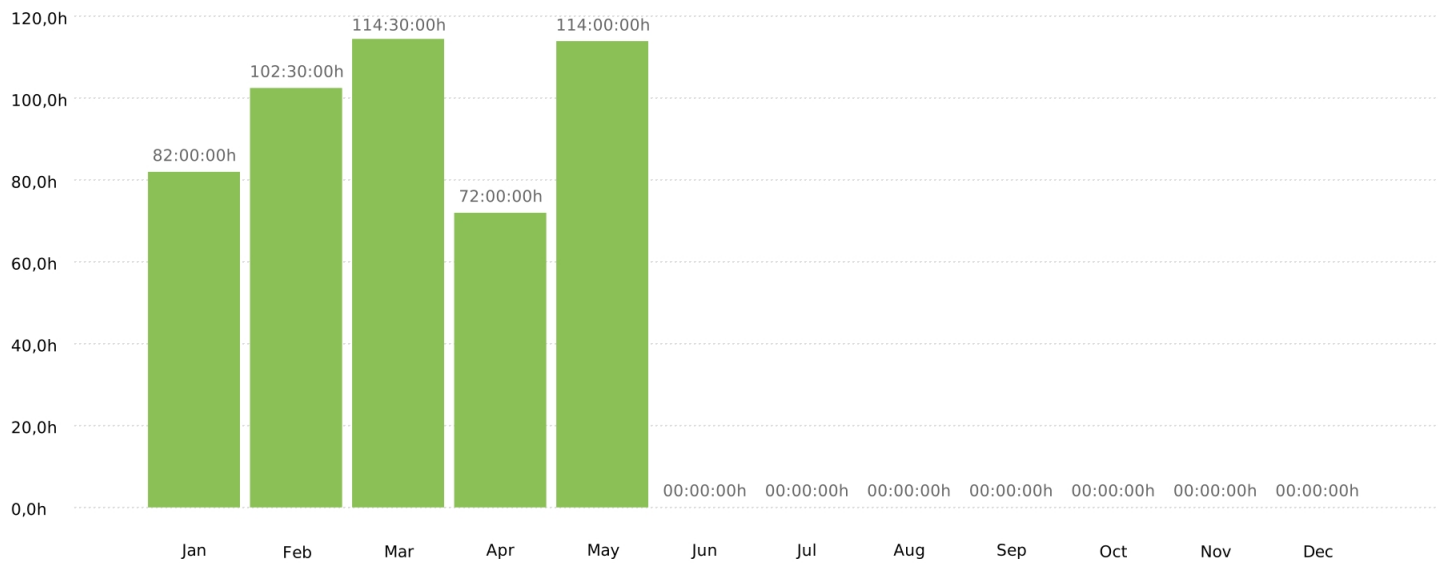
G.5 Wisam Khalid Razi

Summary report

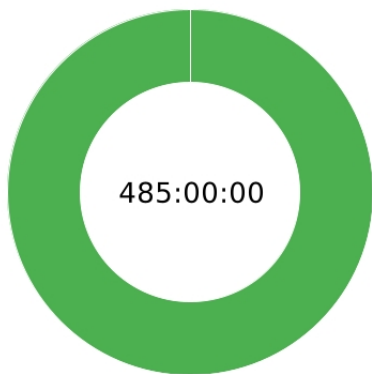
01/01/2023 - 31/12/2023



Total: 485:00:00 Billable: 485:00:00 Amount: 0,00 KR

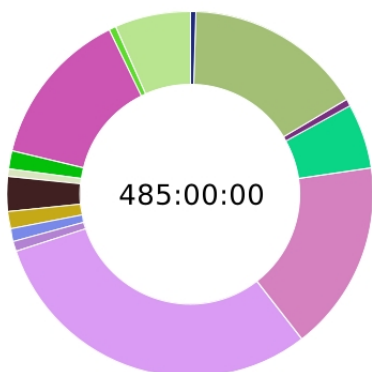


Project



● BachelorTelenor	485:00:00	100,00%
-------------------	-----------	---------

Tag



● Forprosjekt	32:30:00	6,70%
● Forprosjekt, Internt møte	03:00:00	0,62%
● Internt møte	67:30:00	13,92%
● Internt møte, Rapport	07:30:00	1,55%
● Lesing bachelor/master oppgaver	03:30:00	0,72%

● Møte Telenor	14:30:00	2,99%
● Møte Veileder	07:30:00	1,55%
● playbook	05:30:00	1,13%
● Prinsippskisse	04:30:00	0,93%
● Rapport	147:30:00	30,41%
● Research	81:30:00	16,80%
● testing	27:30:00	5,67%
● TestOppsett	03:30:00	0,72%
● Theory	76:30:00	15,77%
● Theory, Research	02:30:00	0,52%

Project / Tag	Duration	Amount
BachelorTelenor	485:00:00	0,00 KR
Forprosjekt	32:30:00	0,00 KR
Forprosjekt, Internt møte	03:00:00	0,00 KR
Internt møte	67:30:00	0,00 KR
Internt møte, Rapport	07:30:00	0,00 KR
Lesing bachelor/master oppgaver	03:30:00	0,00 KR
Møte Telenor	14:30:00	0,00 KR
Møte Veileder	07:30:00	0,00 KR
playbook	05:30:00	0,00 KR
Prinsippskisse	04:30:00	0,00 KR
Rapport	147:30:00	0,00 KR
Research	81:30:00	0,00 KR
testing	27:30:00	0,00 KR
TestOppsett	03:30:00	0,00 KR

Theory	76:30:00	0,00 KR
Theory, Research	02:30:00	0,00 KR

Appendix H

Meeting reports

This appendix contains the meeting reports from the meetings with our supervisor and Telenor during the project.

H.1 Supervisor

meetingWithSupervisor1901

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

January 2023

1. We tell Ernst what our assignment is.
2. Ernst will leave us alone and just advise about the writing.
3. We should have a few chapters of the report completely finished before Easter.
4. Better to ask for a delay on his advise than to deliver to him something not good/finished/ready for advise.
5. The Gant should have some more "deadlines" so that we are able to see if we are behind schedule as early as possible.
6. Always have a caption on tables and images and explain them in the report before the image.
7. There should always be some text between a section header and the first subsection header.
8. It does not hurt to give an impression that we have done more than the assignment asks.

meetingWithSupervisor2601

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

January 2023

1. Alle avgjørelser må ikke være tatt i forprosjektet.
2. Det er en egen privat teamskanal for veiledermøte: "Ernst - supervision".
3. Ernst vil ha møterferat fra veiledermøter.
4. Viser foreløpig (nesten ferdig) prosjektplan
5. Prosjektplanen skal være skrevet på en komplett måte, man skal skjønne hva greia er uten å måtte se på vedlegg.
6. En section burde ikke starte med en liste, burde være et lite avsnitt som forklarer hva listen handler om.
7. Møte uke 6 (09.02) passer ikke til vanlig tid, så det må flyttes evt. avlyses.

meetingWithSupervisor0202

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

February 2023

1. Ernst har ikke fått lest prosjektplanen enda.
2. Noen tekniske utfordringer med utstyret gjør at testingen ikke ser ut til å kunne gjøres helt som vi hadde planlagt.
3. Vi prøver å jobbe med det vi kan mens vi prøver å løse våre tekniske problemer.
4. Vi må huske å ha en rød tråd i teoridelen av rapporten og minne leseren på hvorfor den teorien er relevant til prosjektet.
5. AI verktøy er fult lov å bruke, men da må det skrives om og vi har ansvar for at det faktisk er riktig. Man må være kritisk til hva de sier, for det er ikke nødvendigvis sant.
6. Ernst skal få lest prosjektplanen vår innen mandag og komme med en tilbakemelding.
7. Vi avlyser neste ukes møte, så neste møte blir torsdag 16.02.

meetingWithSupervisor1602

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

February 2023

1. GPS til thingy var veldig upresis.
2. Vi må ta opp problemene våre med oppdragsgiver.
3. Utstyret virker ikke som er egnet for å være i bevegelse.
4. Vi må huske å få dette med i rapporten, hva som ikke funket og hva vi gjorde med det.
5. Hvert 45. sekund virker som et alt for langt interval mellom målinger.
6. Vi bør prøve å gjøre så godt som mulig med det originale utstyret og så får det bare bli mindre bra, også kanskje en parallell løsning som vi kan sammenligne med som vi mener blir bedre.
7. Det viser at vi er uavhengige og fleksible og tenker selv, og det er bra for karakteren på bachelor.
8. Mulig at om vi setter enda lengere tid mellom hver thingy GPS oppdatering så blir den mer presis, men da blir det jo lengere mellom målinger.
9. Utfordringene vi møter bør ikke kunne oppfattes som unnskyldinger for at vi ikke har fått gjort det vi skulle i rapporten.
10. Vi skal være objektive og "gode forskere".
11. Thingy er frustrerende og mye styr.
12. Tåke kan ha hatt noe å si.
13. Problemer å løse = ting å skrive om.
14. Det er kanskje viktig for telenor at det er akkurat dette utstyret som brukes.
15. Huske å sende epost til Ernst når referat er ferdig og lagt ut på Teams.
16. Vi burde få tydelig frem hva som vi har funnet ut fremfor hva som var gitt før oppgaven begynte, Det er viktig for en god karakter.

meetingWithSupervisor2302

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

February 2023

1. Vi har byttet kildestil til den som er i latex malen.
2. Virker som vi kan droppe Thingy.
3. Telenor vil at vi skal fokusere på svingen der det skjer noe med dekningen.
4. Trafikk er kanskje ikke så relevant på vårt testområde.
5. 5G er ikke relevant.
6. Men kanskje skrive et lite avsnitt om 5G i teorien likevel, typ hvorfor vi ikke bruker 5G. Eller kanskje bare nevne i scopet at 5G ikke er relevant for oppgaven.
7. Vi ser på muligheten for å lage vår egen lille app som gjør akkurat det vi trenger.
8. Alternativt bare kjøpe pro-versjon av en eksisterende app for 200kr.
9. Teorikapittellet begynner kanskje å bli litt langt, husk at leser må føle at det er relevant for resten av rapporten.
10. Vanlige oppgaver er 50-70 sider, ikke skrive for mye som ikke er irrelevant. Det skal ikke være langt bare for å være langt.
11. Husk at leser ikke vet noenting om prosjektet. Anbefales å få noen andre som ikke har noen tilknytning til prosjektet å lese gjennom.

meetingWithSupervisor0203

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

March 2023

1. Vi mangler data fra de første testene, men har noe.
2. Vi føler vi ligger bra an, i forhold til fremdriftsplanen.
3. Vi har funnet en god app.
4. Vi har begynt å bruke iperf3, selv om den så langt gir ikke helt de verdiene vi håpet.
5. For å finne superparameteren er det nok lurt å se på de forskjellige parameterene hver for seg og sammenligne for å se hva det er som påvirker dekningen.
6. Båndet kan være viktig.
7. Selv om en test ikke gir noe brukbar data så lærer vi av den likevel.
8. Skriv ned i rapporten hvilken utfordringer vi kom borti under test scenarioene.
9. Noen som leser rapporten og vil gjøre de samme testene burde kunne unngå en del av de feilene vi har gjort, viktig at vi skriver om hva man skal passe på når man utfører testene, f.eks. har man bra nok hardware.

meetingWithSupervisor0903

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

March 2023

1. Ikke noe møte neste Tordag, så neste møte er 23/03.
2. Teorien burde IKKE være 30 sider.
3. Funn: den tviholder på det båndet den allerede er på.
4. Det er betydelig forskjell på hvordan Meraki og smartphone-appen måler verdiene, appen bytter bånd oftere f.eks.
5. Vi må huske å forklare resultat figurene grunding, både hvordan og hvorfor (målingene, farge kodene, grafene).
6. Burde kanskje gjøre fargene i elasticsearch mer forskjellige med tanke på at sensor kan printe ut rapporten, og til og med i svart-hvit. Sensor kan også være fargeblind, så særlig det å skille grønn og blå kan være et problem.
7. Kanskje er det flere parameter vi kan vise fram og som kan være nyttige. Vi kunne fremstille de forskjellige parameterene i forskjellige figurer og diskutere frem superparameteren, og så til slutt lage en "endelig" figur med bruk av superparameteren.
8. Skrive hva som kan forbedres med utstyr og software.
9. Forklare hvor relevante parameterene vi får ut er for ulike use cases. Er båndbredde viktig i usecasen med truckene? Vi kan løfte oppgaven litt ved å diskutere litt rundt hva hvis use-casen var annerledes f.eks.
10. Hvis vi fjerner innrykket i starten av et avsnitt, så kan vi øke avstanden mellom avsnittene. Det er ikke noen klare anbefalinger om hva som er foretrukket av NTNU eller "best".

meetingWithSupervisor2303

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

March 2023

1. Viktigste er å skrive slik at det blir en rød tråd gjennom hele rapporten.
2. Første siden burde hvem som helst skjønne.
3. Teori-kapittelet kan være basert på at leseren har noen lunde samme bakgrunn som oss.
4. Vi kan forvente at sensor har informatikk bakgrunn.
5. Når man kutter ned gjør man det ofte bedre.
6. Man kan kommentere ting ut i latex.
7. Testene bør være kategorisert, men trenger ikke nødvendigvis å være veldig organisert. Må virke passe planlagt.
8. I target audience burde vi skrive hvem rapporten er nyttig for, og på slutten av avsinttet burde vi nevne hva vi forventer av leser for å kunne fortså rapporten, typ folk med informatikkbakgrunn.
9. Hvis figuren inneholder data noen andre har funnet, holder det i utgangspunktet å si hvor det er fra. Trenger ikke å be om tilatelse for slikt som er oftest tilgjengelig.
10. Vi trenger nok ikke å ha et avsnitt om layout og fonter og slikt med mindre vi gjør noe ekstra spesielt og avansert som ikke.
11. Our starting point burde evt inneholde det som er fra universitetet, ikke verdt å ta med ting fra typ vgs. Men det burde ikke være så langt.

meetingWithSupervisor3003

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

March 2023

1. Levere rapport på fredag hvis vi vet vi ikke kommer til å få skrevet noe i helgen.
2. Prinsippskisse kan bli til playbook.
3. Vi må få lest gjennom hva andre i gruppen har skrevet fordi man blir ofte blind for eget arbeid og overser skrivefeil og slikt.

meetingWithSupervisor1304

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

April 2023

1. Vi kan avklare det med bandwidth i introduksjonen.
2. Figur som viser båndene er bare for å friske opp litt. Generelt fint med en figur i ny og ne.
3. Kilden i bånd kan henvises til før punktlisten ved å nevne alle punktene i paragrafen som innleder listen.
4. Så lenge vi gjør det klart at vi fokuserer på 4G så går det fint å nevne 5G en gang iblant siden det er en nyere teknologi og kan hjelpe med å avklare ting.
5. Om prosjektteknisk (slik som "time" i scope og "Team organizing") står i innledning eller ikke kommer nok ikke til å værke trekke opp eller ned. En løsning kan være å ha det, men korte det ned så det tar minst mulig fokus.
6. Vi kan legge alt i kildeliste og ikke ha fotnoter, det er ikke feil og er absolutt enklest. Men veldig små referanser kan være fotnoter, noen liker det, noen liker det ikke.
7. Vi må anta at leseren ikke kjenner til JSON, men hvis vi kaller det et data format, så trenger ikke leser noe mer forklaring enn det.
8. Bachelorpresentasjonene er noe hvem som helst kan komme på. Det at man ser på de andre presentasjonene er en oppfordring, men det er en del som forventer at man ser på de andre presentasjonene i den samme bolken som man selv er i.
9. Presentasjonen skal være om prosjektet, om rapporten tar en sentral rolle eller ikke er ikke så viktig. Så lenge vi presenterer prosjektet som helehet godt.

meetingWithSupervisor2004

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

April 2023

1. Man kan løse det at latex endrer referanser til små bokstaver ved å sette det inn i en ny klammeparantes.
2. Å varsle om en av parameterene er lave kan også fremstilles som en matematisk formel i forhold til superparameteren.
3. Resultater fra tester som er veldig like trenger ikke å nevnes om og om igjen. Bedre da å bare si at de fleste testene ble som denne.
4. Viktig å være sannferdig og si at det er upload hastighet som måles av iperf3.
5. Å ta en test til på download speed er sikkert lurt.

meetingWithSupervisor2704

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

April 2023

1. Meraki dashboardet oppdaterte seg tregt når vi målte downloadspeed, på begge testene.
2. Vi kunne vurdert å ha dashboardet på en annen pc.
3. Scopet bør ikke skrives i fremtid, heller presens/fortid.
4. Eneste gangen det bør skrives i fremtid i rapporten er når man snakker om noe som skal komme senere i rapporten.
5. Ernst trenger tid, helst så mye som mulig til å gå gjennom ”ferdig” rapport.

meetingWithSupervisor1105

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

May 2023

1. Ernst har ikke fått lest enda, vi er nr 2 i køen nå, vi bør få levert en ny, mer komplett versjon raskt, typ fredag 12/05 før kl 14 senest, men hvis vi ser at vi kan gjøre den en del bedre på noen timer kan vi gi bekskjed.
2. Kommentarene kommer da til å være "lettere", mindre grundige enn ved påske, mer på det store bilde.
3. Eneste store endring i kapittel 1-3 er at hypoteser har dukket opp i kapittel 3.
4. Passe på at vi ikke introduserer noe nytt å finne ut av i hypoteser.
5. Datoer og et par andre ting er foreløpig skrevet ganske inkonsekvent, på en del forskjellige måter.
6. Bra å skrive hvorfor resultater styrker/svekker våre hypoteser.
7. Bra å vise frem de testene som er unntak fra normalen, og et eksempel på hva som er normalen.
8. Sikkert en god ide å la Henning og co. få se rapporten, så kommer nok de til å fokusere på introduksjon i sine evt tilbakemeldinger.
9. Ernst sitt inntrykk er at vi har kontroll.
10. Ernst vi prøve å holde av noen tider etter kl 12 på fredag 19/05 til eventuelle små digitale møter om det viser seg å være nødvendig.

meetingWithSupervisor1905

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

May 2023

1. ~forran cite gjør at det blir mellomrom, men ikke linjeskift før kilden.
2. Er nok lurt å gjenta figurer så man slipper å bla veldig mange sider tilbake.
3. Det viktigste ifølge Ernst er å korte ned litt og gjøre det mer konsist.
4. Hypoteser er ikke noen krise egt, men det er nok bedre om man gjør hver hypotese helt konkret, med en setning. En annen variant er å kalle det "forventninger" for å gjøre det mindre stivt. Vi kan også si at vi "myker opp" begrepet hypotese litt slik at lite trenger å endres.
5. Viktig at konklusjonen "oppsumerer" oppgaven og ikke kommer med noe nytt. Man skal kunne lese innledningen, så konklusjonen rett etterpå og skjønne at de henger sammen.
6. Fint å forklare store tabeller og figurer og hva som er poenget med dem før den kommer.
7. Ingen vits i å forklare skriptene.
8. Vi kan vurdere å kutte en del ned på de tingene som ikke endte opp med å bli brukt noe særlig.
9. Vi kan bare nevne i notater i introduksjonen at alle figurer der det ikke står noe annet er selvlaget med denne CC.
10. Figurer som er inspirert av kilder er det bare å skrive "inspired by ...".
11. Kan hende at det ikke er opplagt nok at hver dag består av test 1, 2 og 3.
12. Ingen preferanse på måte å skrive datoer på.
13. Hvilke begreper som passer i glossary er opp til oss.
14. IE og IIK er vårt institutt og fakultet.
15. Vurder kommentarene med et kritisk blikk.

16. Helhetsinntrykket er at det er en bra rapport, men det kan selvfølgelig bli litt bedre.
17. Presentasjonene har typisk vært rundt 15 minutter og 5-10 minutter med spørsmål.
18. Det er bare å kontakte Ernst på epost eller telefon om det er noe på slutten.

H.2 Telenor

meetingWithClient2601

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

January 2023

1. Bjørn kan hjelpe med Nordic-thingy programering og ledning. Han skal undersøke og komme tilbake.
2. Foreslår å pakke Meraki inn i noe som forstyrrer signalet veldig og se hvor lang tid det tar før utslaget blir forstyrret. Sånn kan vi finne hvor forsinket signalet er. F.eks. en heis. Vi kan også ha en spike i data-trafikk og se hvor lang tid det tar før det får utslag, men det kan være en separat forsinkelse.
3. Bjørn skal på 4 timers Meraki-kurs og kan kanskje dele noe dokumentasjon derifra.
4. Et underpunkt på scope om teori er kanskje lurt.
5. Kanskje knytte scope nærmere use-casene som fremgår av innledningen. Hele tiden referere til use-casen(e) gjennom rapporten (og forprosjektet).
6. Kanskje inkludere litt mer spesifikt om testing i scope, som hvor mange turer vi kjører per test, at vi trenger flere observasjoner.
7. Henning vil gjerne ha en figur som viser testoppsettet i rapporten innen neste møte. Gjerne flere figurer.
8. Visio \rightarrow svg-format \rightarrow vektorbilder
9. På neste møte kan vi diskutere mer spesifikt om testscenario.
10. Husk på anvendelsen når vi skriver teori, skriv teori om hvorfor vi tester som vi gjør, hvorfor vi kjører frem og tilbake, hvorfor vi kjører sakte, hvorfor Merakien står som den gjør osv.
11. Lov å komme med litt hypoteser i rapporten, "vi tror ...".
12. Hente inspirasjon fra Andras' bachelor.
13. Skrive hvorfor vi tenker å teste slik, hva er hypotesen og teorien bak det.
14. Kanskje lurt å visuelt følge med under en kjøretur for å se hvilket bånd den kjører på.

15. I teori-kapitlet kan vi inkludere hvor mye kapasitet Telenor har i 1800 båndet og hvor mye er i 800 båndet.
16. Neste møte Torsdag 09.02 10-11, ha klar figur, mer teori og demo Meraki.

meetingWithClient0902

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

February 2023

1. Bjørn la ut link til Nordic for å stille spørsmål om Thingy
2. Thingy bruker en stund på å få en posisjon
3. Si ifra til Bjørn hvis vi ikke får Thingy til å gjøre som den skal innen rimelig tid
4. Henning savner data plan, i skissen, en 4G linje ut til internett fra Meraki og teams på internett. Linjen til Meraki skyen kan hete kontrollplan.
5. Teamsmøte burde være med en klient som er på et stabilt sted med bra kvalitet, f.eks. på campus med kablet internett.
6. En skisse til hvor vi legger til Teams
7. Det at merakien bare kan oppdatere seg hvert 45.sekund er et funn vi burde beskrive.
8. Merakien er ikke designet for å være i noe som beveger seg.
9. Henning foreslår at vi har flere testscenarier, en hvor vi kjører og stopper og en annen hvor vi kjører kontinuerlig. Kanskje også flere testscenarier med forskjellig hastighet.
10. Vi kan også bruke en smartphone til å sjekke dekningen (og evt posisjon)
11. Vi burde treffe alle fargekodene i heatmappet, et par målinger per fargekode ideelt sett.
12. Den foreslåtte strekningen er 1.1 km.
13. Henning foreslår at vi tar en tur i uke 7 (13/02-19/02) der vi sjekker dekning med mobilen, rekonosierer og tar litt bilder.
14. prosjektmøte 3 blir 22.02 med scoping av test som hovedtema
15. Fokus frem til neste møte skal være tester
16. Kart med basestasjoner skal Bjørn legge ut til oss. Henning vil ha det tatt i bruk i teorikapitlet.

meetingWithClient2202

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

February 2023

1. Få skrevet om hvordan Thingy oppfører seg i rapport.
2. Lurt å starte målingene der det er 1800 MHz bånd.
3. Henning vil ha når vi er på 800- og 1800 MHz båndet.
4. bånd B3 er 1800 MHz bånd og bånd B20 er 800 MHz bånd.
5. Hvor bra api-et er for oppgaven er et funn å skrive om (den har vanskelig for å vise hvilket bånd som brukes)
6. Dokumenter når båndene skifter og at det er i sammenhengen med deknin-gen fra hver basestasjonen. Fortelle at brukeropplevelsen går ned men ikke stryken på radio signalet.
7. viktig at vi får med ALLE data fra testscenarioene, også selvbestemte parametere som bilens hastighet, antall gjennomkjøringer, hvor utstyret er osv.
8. Manuell test for å se hvilken data vi får ut.
9. Prøve å sammenligne tems målingene med våre målingen.
10. Ide å kjøre kontinuerlig speedtest som vi kan legge oppå som enda en måling. Den vil vise tydelig når vi bytter bånd og basestasjon.
11. Viktigste blir å dokumentere akkurat der kvaliteten endres så kanskje en ide å stoppe akkurat rundt der. Stå flere steder i den ene svingen for å få nøyaktig data der det er dårlig dekning. Dokumentere kvalitets endringen.
12. Viktig å få frem hvorfor vi gjør testene som vi gjør, hvorfor den hastigheter på kjøringen, hvorfor vi stopper der vi stopper osv.
13. Teorikappittel på hva som skjer når man bytter mellom båndene.
14. Noe av det viktigste er hvilket bånd vi er på.
15. Bjørn skal sende over en speedtest som sender data mellom to pc-er og tester hastigheten kontinuerlig. IPERF3.

16. Begynn å se på hvordan presentere alle parameterene på en måte som blir mest leselig og best enklest for leser å henge med. Finne en bra representasjon av datafunn våre som gjør funnene oversiktlig i elasticsearch.
17. Flere lag i elasticSearch som viser bånd, deknings-hastighet og db-målinger. db målinger er ikke bra nok for å måle kvalitet. Hvilke parametere har størst innvirkning på brukeropplevelsen? Universell kpi for kvaliteten av data.
18. Ikke bruk for mye tid på å få ut signaldata fra Thingy, eller Thingy generelt da GPS posisjon kan fås fra annen app.
19. Henning foreslår å droppe Thingy og erstatte den med en app.
20. Henning vil ikke at vi skal bruke for mye tid på å få dekningsdata fra mobilen, og la den fokusere på GPS posisjon.
21. Få gjort målingen og beynn å skriv.
22. Gjør en test og skriv en hel rapport fra introduksjon til konklusjon om den ene testen. Da ser vi hvordan det blir og om vi evt mangler noe.
23. Møte onsdag 8.mars kl 12
24. Fokus på å få test-dataene strukturert

meetingWithClient0803

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

March 2023

1. Den aller viktigste tingen å dokumentere er at det er mindre kapasitet på 800 båndet enn på 1800 båndet i praksis.
2. Gjerne send den delen over til Telenor for gjennomlesning.
3. Vinduet som sier hvilket bånd vi er på oppdaterer seg raskere enn resten av dashbordet.
4. "radiohygiene" dropper ikke så mye når den går fra 1800 til 800 båndet, men dropper før den bytter til 800 fra 1800.
5. Vi må samle "bevis" for å bygge opp under teorien.
6. Viktig at vi forklarer hvordan vi legger parameterene sammen for å lage en fin presentabel grafisk fremstilling av dataene.
7. Vi må nok legge med script i rapporten.
8. Viktig å få med at vi har endret "systemarkitekturen" underveis (benytter apper istedenfor Thingy).
9. Kanskje lurt å skrive et avsnitt om hvordan vi kvalitetsikkrer dataene vi får, at vi får vist at vi har tenkt på det.
10. Viktig at sensor får inntrykk av at dataene er korrekte og kvalitetsikret (ved å gjøre mange tester).
11. Få frem at på grunn av radiohygiene er det ikke nødvendigvis så svart-hvitt at 800 båndet er tregere enn 1800 båndet.
12. Kanskje fremstille båndene som rør hvor dataen kommer gjennom, hvor da 1800 båndet har et rør med større diameter enn 800, og radiohygiene er at røret blir klemt på.
13. Henning er fornøyd med hvordan datene blir fremstilt nå på elasticsearch.
14. Vi kunne kanskje flytte startpunktet for å kutte ned testtiden.
15. Kunne vært en ide å vise hvor den bytter bånd i elasticsearch.

16. Henning savner å kunne se på kartet i elasticsearch hvor båndene er og hvordan hastigheten er. F.eks. med en rounding i forskjellig størrelse avhengig av bånd.
17. I diskusjonen burde vi diskutere hvorfor den bytter til 1800 båndet når vi kjører nedover lengere nede enn den beholder 1800 båndet når vi kjører oppover.
18. Henning er fornøyd med fremdriften og mener at vi kanskje ikke trenger så mange flere målinger.
19. Kanskje skrive litt om at et mobilnett ikke bare er et mobilnett, at det har blitt mer komplisert enn det med stadig høyere krav til funksjonalitet og kapasitet.
20. Kanskje dele inn dataene i kjøreretning A og kjøreretning B.
21. Den tviholder på det båndet den allerede er på uavhengig av hvilket bånd det er.
22. Bare å mase om det er noe vi trenger å vite eller få avklart.
23. Til neste gang systemskisse/prinsippskisse og teorikapittel og bilder fra.
24. Systemskisse: Powerpoint som viser testoppsettet og forklarer helheten i testen + litt grafisk fremstilte resultater. Hva det forskjellige hardwaret gjør og hvordan dataen blir prosessert. Den skal være fin å se på.
25. Neste møte 30/03.

meetingWithClient3003

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

March 2023

1. Få med i rapporten at Merakien er laget for å være stasjonær.
2. Bevis funn med resultater.
3. Hvis vi får god tid kunne vi tenkt på hvordan vi skulle lage en demo med Elastic search med playbooken på Teams. Burde da tenke litt pedagogisk.
4. Software section i prinsippskissen/rapporten.
5. Legg gjerne ut info om det er mulig å komme på presentasjonen på Telenor teams kanalen.
6. Når kan det passe med pizza og øl, hvilken dato? 25. mai?
7. Kan ta med han vi snakket med som måket som en liten historie om en fyr som er lei av dårlig 4G dekning i rapport og presentasjon.
8. Koble sammen bildene i slide 3 og 4 som forklarer test oppsettet.
9. Kanskje skille Teams fra alt annet da det handler mer om brukeropplevelsen istedenfor det tekniske. Forklare brukeropplevelsen vår via 4G tilkoblingen.
10. Skissen på data processing burde kanskje si noe om hvilke parametere som hentes fra de forskjellige loggene og hvorfor. Få fram tydelig hva vi logger fra hvilken enhet, data processing skissen burde være mer detaljert.
11. Generelt forklare skissene sånn at hvem som helst kan forstå dem.
12. Bruk bedre figurer for loggene, f.eks. lyn symbol for iperf.
13. Prøv å illustrere med så få ord så mulig, men heller fokuser på bruken av ikoner, siden de gir en bedre forklaring.
14. Kanskje lage noen hypoteser på hvordan Merakien kommer til å oppføre seg, som at den kanskje henger lengere på båndene enn mobilen gjør.
15. Knytte sammen RSRP og hastighet med bånd skiftet.

16. Bruk skjermbilder fra Elastic search i rapporten er lurt.
17. Henning vil ha med mailen til politiet i presentasjonen. Kanskje også ta med litt morsom statistikk om hvor mye vi har kjørt i så sakte tempo.
18. Skriv i konklusjonen om MG21 er egnet for tip-truckene eller ikke, evt hva MG21 heller bør brukes til.
19. Til neste gang: vise oppdatert figurer i presentasjonen/playbooken/demoen.
20. Neste møte blir mandag 24/04.

meetingWithClient2404

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

April 2023

1. Playbooken kan være en liten x-faktor som vi kan vise til sensor som kan løfte oppgaven.
2. Henning foreslår å ha en video av demoen.
3. Henning vil ha et ikon for dekning på Merakien på figuren, forklare RSRP og RSRQ og slikt med litt mer forståelige begreper.
4. Kanskje ha en figur som viser en mobilmast og merakien som beveger seg, eller på en annen måte være tydelige på at merakien beveger seg fra A til B.
5. Bytte ut download-symbolet med et symbol mer som et speedometer.
6. Mobilikonet burde også ha et ikon for dekning, samme som merakien.
7. Vise et kart som viser hvor vi kjører.
8. Henning tenker: første bilde er kartet, andre bilde er dataprossessering, tredje bilde er noen interessante resultater. Så kommer elastic demo.
9. Playbooken skal være slik at en som ikke vet noe om det skal kunne holde en demo med hjelp av playbooken. En video kunne vært en fin backup løsning.
10. Kanskje test playbooken/demoen ved at noen later som de er kunde som ikke vet noe særlig om nettverk.
11. Playbooken skal hjelpe noen som kan litt nettverk til å presentere og forklare noen som ikke kan noe om nettverk.
12. Større fokus på pedagogikk og bruke symboler som de som ikke kan noe om nettverk kjenner igjen og forstår.
13. Vi starter med et kundeproblem, så bryter vi det opp i forskjellige deler og parametere, så drar vi de dataene gjennom en prossess som ender i elastic search, og til slutt kommer det ut et nyttig resultat i elastic search (ved hjelp av superparameter).

14. Henning skal få lagt ut noen symboler vi kan bruke på teams-kanalen.
15. Vi har nå tenkt ingredienser, men playbooken vil også ha prosessen for å få ferdig produktet (stort fokus på prosessen, skal brukes til å lage demo(video)).
16. Elastic demoen skal tas opp.
17. Å tagge Henning og Bjørn i Teams hjelper veldig på at de får sett det fort.
18. Ting bør være mer forenklet i sammendraget, uten så mye fagbegreper.
19. Kunden skal få noe så enkelt som en fargekode som er grønn om dekingen er bra, oransj hvis den er ok og rød om den er dårlig.
20. Nytt møte torsdag 04.05.
21. Da vil henning ha litt pedagogisk intro klar (background fra rapport eller sammendrag?). Gjennomgang av figurer og playbook. Også helst en liten video av starten av demoen.

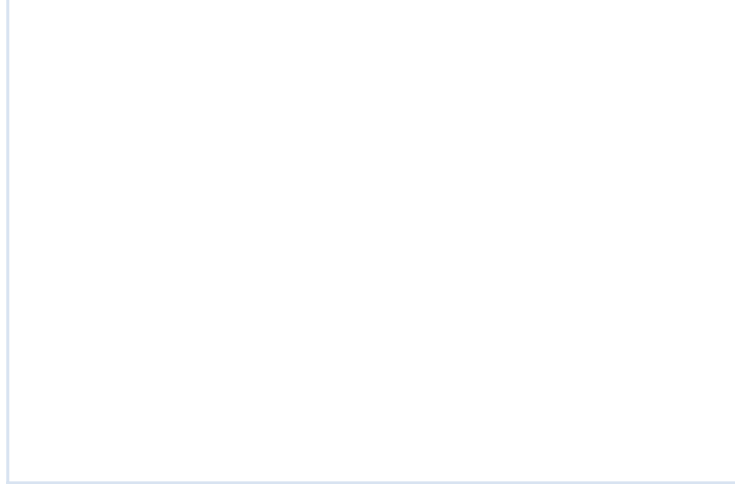
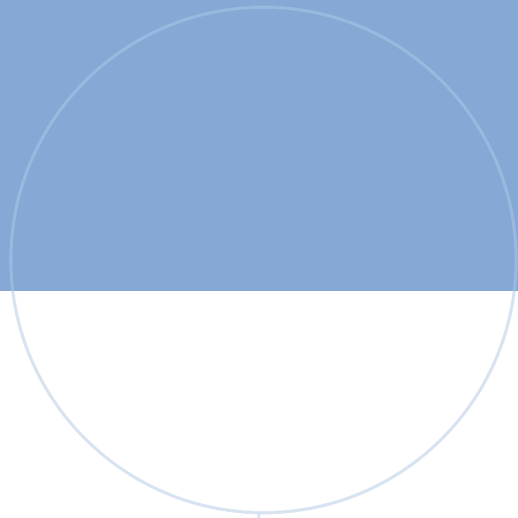
meetingWithClient0405

Lars Martin Nygaard Espen-Andreas Oseth
Vebjørn Albinson Wisam Khalid Razi

May 2023

1. 2 figurer, en som er mer teknisk og en som er mer ment for kunden, for å være enklest mulig.
2. Den enkle burde ikke ha noen fagbegreper, heller mer daglige ord som download speed, signal strength osv.
3. Bruk screenshots fra elastic i figuren istedetfor graph- og heatmapsymbol.
4. Innledningen (playbook/presentasjon) må være enkel som hvem som helst må forstå (også kanskje i rapporten).
5. For å få det perfekt burde symbolene matche mer, men det er VELDIG pirkete
6. Punktene over er forslag, ikke krav.
7. heatmap er bra "alle" skjønner heatmap.
8. Henning vil lage en alternativ presentasjon, et forslag, og legge det ut til oss i Teams.
9. Henning foreslår å lage en veldig enkel figur til innledningen (background), kanskje samme som første figuren i presentasjonen.
10. Veldig viktig å få frem motivasjonen for oppgaven i innledningen, hva oppgaven kan brukes til.
11. Bruksområder: Romarheim (steinbrudd), roboxi (flyplass).
12. Henning er bra fornøyd med background i rapporten, men vil ha figur, prøv å få inn litt hvordan det fungerer med at basestasjonene er faste punkt og enheter beveger seg rundt dem. Det er finpuss.
13. Kvaliteten på konnektiviteten blir bare mer og mer kritisk, og da også å kunne dokumentere kvaliteten.
14. I konklusjonen bør det stå litt om anbefalinger, om dette er en god løsning eller hva som vi tror evt har potensialet for å kunne blitt en bedre løsning.

15. Videoen er akkurat det Henning ville ha, (men med lyd selvfølgelig)
16. Vi bør ikke bruke noe særlig mer tid på å finpusse video.
17. Dette var siste planlagte møte før innlevering.



 **NTNU**

Norwegian University of
Science and Technology