TELCO RECOMMENDATION BASED ON SPEED TEST DATA

By

CHIA ZHEN YIK

A REPORT

SUBMITTED TO

Universiti Tunku Abdul Rahman

in partial fulfillment of the requirements

for the degree of

BACHELOR OF INFORMATION TECHNOLOGY (HONOURS)

COMMUNICATIONS AND NETWORKING

Faculty of Information and Communication Technology
(Kampar Campus)

JAN 2023

UNIVERSITI TUNKU ABDUL RAHMAN

REPORT STATUS DECLARATION FORM

Title: Teclo Recommendation Based on Speed Test Data

Academic Session: JAN 2023

I CHIA ZHEN YIK

(CAPITAL LETTER)

declare that I allow this Final Year Project Report to be kept in

Universiti Tunku Abdul Rahman Library subject to the regulations as follows:

1. The dissertation is a property of the Library.
2. The Library is allowed to make copies of this dissertation for academic purposes.

Verified by,

(Author's signature) (Supervisor's signature)

Address:

No 126, Lorong 11,

Taman Seri Iskandar. 36400

Hutan Melintang, Perak

Aun Yichiet

Supervisor's name

Date: 21/4/2023

Date: 24/4/2023

Universiti Tunku Abdul Rahman			
Form Title : **Sample of Submission Sheet for FYP/Dissertation/Thesis**			
Form Number: **FM-IAD-004**	Rev No.: **0**	Effective Date: **21 JUNE 2011**	Page No.: **1 of 1**

Faculty of Information and Communication Technology

UNIVERSITI TUNKU ABDUL RAHMAN

Date: 21/4/2023

SUBMISSION OF FINAL YEAR PROJECT /DISSERTATION/THESIS

It is hereby certified that Chia Zhen Yik (ID No: 19ACB04782) has completed this final year project/ dissertation/ thesis* entitled "Teclo Recommendation Based on Speed Test Data" under the supervision of Dr. Aun Yichiet (Supervisor) from the Department of Faculty of Information and Communication Technology (FICT).

I understand that University will upload softcopy of my final year project / dissertation/ thesis* in pdf format into UTAR Institutional Repository, which may be made accessible to UTAR community and public.

Yours truly,

____

(*Chia Zhen Yik*)

DECLARATION OF ORIGINALITY

I declare that this report entitled "**TELCO RECOMMENDATION BASED ON SPEED TEST DATA**" is my own work except as cited in the references. The report has not been accepted for any degree and is not being submitted concurrently in candidature for any degree or other award.

Signature : ______________________________

Name : CHIA ZHEN YIK

Date : 21/04/2023

ACKNOWLEDGEMENTS

I would like to express my sincere thanks and appreciation to my supervisors, Dr.Aun YiChiet who has given me this bright opportunity to engage in this telco recommendation system project. It is my first step to establish a good project. A million thanks to you.

Furthermore, I would want to thank my parents for assisting me in shaping my life with passion and a wholly positive approach. Without them, I would not have been able to complete the assignment with a good attitude. Finally, I would want to thank all of my friends that assisted me and were eager to offer wise comments to improve the quality of my work.

ABSTRACT

The project focuses on the telco recommendation using collected data in Kampar hot spot area such as the hostel for the student and Universiti Tunku Abdul Rahman Kampar Campus. And this project is only used for academic purpose. It will provide students with data sciences analysis skill and knowledge, as well as networking knowledge. The telecom industry has vast volumes of data, maybe more than any other institution on the planet. Capturing the insights from this unstructured dataset and generating business insights using data science skills is the battleground. It is one thing to have useful data obtained in a traditional fashion; it is quite another to translate that data to a commercial use case and make your firm millions of dollars. Today's telecom administrator's web page has an enormous number of options, making it increasingly difficult for a customer to choose his or her optimal item selection. Thus, this project aims to develop and carry out a telco recommendation system using collect data to be satisfied people today. Furthermore, the main function of this proposed system is to develop a systematic method that improves existing cellular network data collection and reporting, helping the world become more connected via cellular networks. Moreover, some telco has better service quality depending on the locations due to the network operator investing strategically in the different geographical areas to cater to different demographics. Hence, choosing the right telco becomes a dilemma. Thus, the project aims to develop a system to convenient today the in Kampar area, which recommends them a that has a better network speed test result based on the user location. Other than that, the dataset in this project will be implemented by data pre-pre-processing to let it become more precise, provide more exact data to the users, and also allow other people or companies to obtain data from our database, to lead the world become more connected via cellular networks. Besides that, the system will perform a calculation to the data from the collected data and show to the user such as providing the average of download scores, upload scores, and latency scores. So that, it will improve the accuracy of data to the user who is finding the accurate data on the network. Lastly, this project will provide the data visualization for the user, let the user can easily view all the data in each location in the Kampar area.

Table of Contents

LIST OF FIGURES

LIST OF TABLES

LIST OF ABBREVIATIONS

RSSI	Received Signal Strength Indication
VPN	Virtual Private Network
NaN	Not a Number

CHAPTER 1 PROJECT BACKGROUND

1.1 Problem Statement and Motivation

With the progress of the times and the development of science and technology, WI-FI is becoming more and more important for people nowadays, after leaving the WI-FI area, we will lose the network, so the mobile data has gradually become a must-have for smartphones nowadays. However, the mobile data are not, sometimes it can perform well, provide us a high-performance network, but sometimes it is unstable when the signal are not stable. And the network signal is provided by the telco, the telco cannot measure each region can have a stable and high-performance network coverage. This is because the marketing of telco will do the research and choose a strategy location that can maximize profit to construct the signal station. So that, I would like to develop a system about the telco recommendation using collected data for the user who are confusing to select the telco company and also have a few problems that have affect my daily life:

1) These data are collected from real world user tests who are most likely experiencing network issues while performing the test

This is an era of ever-changing technology sweeping the globe, the expectation of people today is become higher day by day. There are many telcos in Malaysia, such as Maxis Communications, Celcom (Axiata), Digi Communications, U Mobile Sdn Bhd and so on. However, the first thing that will be done by the user when they are facing the network latency is network performance testing, such behaviour will affect the data. So that, the existing signal data is not a comprehensive representation of the overall cellular network quality and is mostly biased towards network congestions and peak hours. This is because most of the people will not be using mobile data at the midnight, the data will just show that it is unstable when doing network performance testing during peak hours such as 6 p.m. and 11 p.m. on weekdays [1]. Therefore, people today need to spend more time to find a good telco company which have a better network performance, and most of the time the information find by themselves is not accurate because the data will be disturbing by the people who like to use speed test when peak hours.

2) Existing tools like open signal and nPerf has scattered speed test spot that do cover less accessible areas

Nowadays, most of the people like to use the speed test system when they are facing problem, and most of them are test when they are in the location that sure will not have a good signal reception. Barriers to geographical reception are frequently insurmountable. The construction materials utilized in our house or workplace, or damaging interference from the structures surrounding you, are significantly more likely to be the source of our poor mobile signal. Within home's walls, cellular signals have a hard time travelling through metal and concrete. That's why hanging out of an upstairs window can provide us with two or three bars of service. Cell signals will be reflected off buildings and other obstacles in cities, resulting in many signals moving in various directions. These many signals will interact with one another, and some of them may lose power, resulting in a reduced cell signal for you. Therefore, all the situations in the above will affect the data become not accuracy.

3) Existing tools only use network bandwidth as the metric for service quality, rather than latency and jitter scores

Most of the online speed test system are using the network bandwidth as the metric for service quality, rather than latency and jitter scores. However, a metric for a network is not only the bandwidth. This is because jitter can result in the timing drift, network congestion or route changes such as the quality of audio and video transmission might suffer when there is too much jitter [2]. Following that, latency is a term used to describe network communication delays. Latency is described in networking as the amount of time it takes for a data packet to be collected, transported, processed by many devices, then received and decoded at its destination [2]. Hence, not only the bandwidth is the metric for the network quality, but most of the online speed test system are use the network bandwidth as the metric, it will lead the user think that the higher the bandwidth, the smoother the network, the importance of the latency, jitter, and reliability scores will also affect the network experiences.

1.2 Project objective

In this project, I will develop a system for recommend the user the telco which have a best network speed based on their location. And using the package provided by each telco which is non-restricted on the network speed cap to collect the network speed test data.

1) **To design a data collection method to obtain accurate network quality and speed test data.**

In order to perform this project well, the existing system will be compared and studied in order to provide this project an enhanced and comprehensive functions to achieve the user expectation toward the telco recommendation system. So that, in this system will use the research result from the literature review to have a better recommendation method for the use which is this system will use the Download Speed, Upload Speed, and latency be the metric for the recommendation method for the user. In order to have an accurate data to avoid unfair comparison. Other than that, I will use the package provide by the telco which have not any network speed cap to reduce the unfair comparison [3]. After that, I will utilize the signal data from the collected data to develop the more accurate data from each operator, I will optimize the data become more precise, provide more exactness data to the users, and also allowing other people or company to obtain data from our database, to lead the world become more connected via cellular networks. Besides that, the system will perform calculation to the data from the collected data and show to the user such as provide the download scores, upload scores, latency scores. So that, it will improve the accuracy of data to the user who are find the accurate data on the network.

2) **To sanitize the dataset using data pre-processing techniques to improve dataset quality.**

In this project, the data will perform data pre-processing techniques to improve dataset quality. When I was collecting the data, there may occurs some error on the dataset such as the download or upload size may occurs NaN values, it may affect the data balancing. In order to solve this problem, I will use the data pre-processing to improve the dataset quality. Firstly, I will determine which part may have the NaN value, and also delete the part that will not affect the dataset quality such as the server location, InternalIp, and ExternalIp. After that, I will use the Fill NaN method in the pre-processing method to fill the Zero instead of the NaN value to improve the dataset quality. In this way, the system will provide users with the ability to view the more precise data and the system have more ability to do recommendation for the user.

3) **To develop a recommendation system using the network quality data for location-aware telco recommendation**

Not only to visualize the network quality metrics on a map, but this system will also have a recommendation system using the network quality data for location-aware telco recommendation once the quality data for various locations has been collected in the excel file. This system will retrieve the current geographical coordinate from the user and based on the user's current geographical coordinate to do telco recommendation, let them not need to waste the time to research for the performance of the local telco and easy to know the strength and weakness for each local telco.

4) **To pursue the latest data set, host an online speed test system and a dedicated database is compulsory.**

Another objective of this project is to host a comprehensive network speed test system that accurately measures the upload and download speeds, latency, and other metrics of users' internet connections. The objective of this project is to enhance the user experience and network performance of mobile network users through the development of a comprehensive network speed test system for future work. Due to the rapidly changing nature of mobile network technology, data collection by alone methods may not provide enough data or be insufficiently to keep up with the latest developments. Other than that, the technology is constantly evolving, which can impact the accuracy of speed test data and Commercial factors, such as the investment required to upgrade network infrastructure, can also impact the timeline for new technology rollout [4]. These factors can lead to discrepancies between collected data and actual network speeds for today. To address this issue, the project aims to host a custom speed test website on a local laptop to collect data from various mobile network providers. And the dedicated database can protect the user's privacy.

1.3 Project Scope

This project's scope is covering a systematic method that improves existing cellular network data collection and reporting. Cellular network is the radio network which is dispersed over land regions and known as cells, which are provided by at least one fixed-location transceiver known as base station. This is because the ways of existing cellular network data collection will affect the data accuracy. And it can lead the world become more connected via cellular networks after improving the existing cellular network data collection and reporting.

Besides that, Geographic reception limitations are frequently insurmountable. The materials used in our home or business, as well as harmful interference from the structures around you, are far more likely to be the source of our bad cell signal. ISP invest strategically in the different geographical areas to cater to different demographics; some telco has better service quality depending on the locations. Therefore, with the progress of the times and the development of science and technology, telco is becoming more and more important for people nowadays, after leaving the WI-FI area, mobile phone will lose the network, so the mobile data has gradually become a must-have for smartphones nowadays, and users today are choosing the right telco becomes a dilemma. So that, this project aims to develop a universal cellular network service quality monitoring tool to recommend users which are the best telco based on their location in Kampar area.

Furthermore, this project will collect the cellular network data by using collected data and use the algorithm to calculate the best network in each location such as The Trails of Kampar, Westlake Villas Condominium, Kampus West City Condominium and so on. In this project will use 5 telcos to do the comparison in each location such as Umobile, Digi, Maxis (Hotlink), Celcom and Yes. In order to get a accurate data, the package provided by each telco is critical. I will choose the unlimited speed cap for each package provide by the telco to avoid unfair comparison.

1.4 Contributions

In the project, the contribution is to provide an application for the people today easily and efficiency to find a best telco based on their location by using collected data. Besides that, the application will provide a complete dataset about the network metric on Kampar area such as the download and upload speed, download and upload size, and latency. In order to obtain a better quality of the dataset, I will sanitize the dataset using data preprocessing techniques to improve dataset quality and make it open source on the Flask-based systems, let the person who is needed to view and utilize the dataset to lead the world become more connected via cellular networks. Other than that, in order to provide an efficiency system for the user, this system provide a map for visualize the network quality metrics for the user can easily to view the best telco on the map according to the network metric from the dataset. Lastly, this system will recommend the best telco to the user by using the network quality data for location-aware telco recommendation.

1.5 Report Organization

This report is divided into five chapters, each of which explains a distinct aspect of the project. In Chapter 1, it will cover the problem statements and motivation, project objective, project scope, and contribution. In Chapter 2, it is a section for literature review. Three mobile applications with comparable themes are reviewed. These projects' functionality is described along with their corresponding strengths and weaknesses, and finally provide the limitation solution for each application. In Chapter 3, This report will cover the system design, requirements, timeline, and estimated cost, as well as the system design, overall system flowchart, functional modules in the system, algorithm design, and graphical user interface design in Chapter 4. Furthermore, it will also discuss the hardware and software setup, system operation in Chapter 5, and testing results, as well as the challenges faced during the project and the evaluation of the project objectives in Chapter 6. Finally, in the Chapter 7 will conclude with some remarks on the project's significance and potential future improvements.

CHAPTER 2 LITERATURE REVIEW

2.1 Literature Review

Cellular networks are high-capacity, high-speed voice and data communication networks that enable cellular devices by improving multimedia and allowing for seamless roaming. As mobile devices become more ubiquitous, these networks are being used for more than just entertainment and phone calls. They have become the primary form of communication for financial-sensitive corporate transactions, life-saving situations, and life-critical services such as E-911[5]. Today, these networks are the lifeblood of communication. Mobile base station antennas connect with your mobile device, which then sends a radio signal to the antenna, similar to how a two-way radio works. This signal is then transmitted down the tower to a cabinet at the bottom, where electrical equipment processes it and determines where it should be transmitted. Topography, physical restrictions such as trees and buildings, the estimated number of calls in the cell, and the radio frequency at which the base station will operate, often known as "spectrum," all impact the location of mobile base stations [6].

After that, a mobile phone signal is the intensity of a signal received by a phone from a cellular network. The strength of the signal will vary based on a variety of factors such as proximity to a tower, any barriers like as buildings or trees, and so on. Most mobile devices display the approximate strength of the received signal as a sequence of bars of increasing height [7]. For the example, cellular signals have a difficult time travelling through metal and concrete within a home's walls. That's why we can get two or three bars of service by hanging out an upper window. Cell signals in cities will be reflected off buildings and other obstructions, resulting in multiple signals travelling in different directions. These numerous signals will interact, and some of them may lose power, resulting in a lower cell signal for you [8].

Moreover, we will use the Opensignal to do the data collection, this is because billions of individual measurements are collected from over 100 million devices globally by Opensignal. Opensignal collects data every day of the week, at all hours, and in all areas where people live, work, and travel: there are no models, projections, or idealized testing circumstances. the data originates from actual smartphone users, and report on their actual network experience, whether they are indoors or outside, in a congested metropolis or walking through the countryside [9]. Therefore, the data will be more accurate, but due to the data that show by the Opensignal is more focused on bandwidth, so we will analysis the latency and reliability scores for the network.

When we talk about the metric of network performance, not only the bandwidth, but it also has the throughput, latency, signal strength, and reliability. In bandwidth, it is defined as the amount of data or information that may be sent in a certain amount of time, and high bandwidths do not imply faster performance, and the quantity of messages successfully transmitted per unit of time is referred to as throughput. It is determined by available bandwidth, signal-to-noise ratio, and hardware limits [10]. After that, latency is a unit of measurement for delay. In a network, latency is the amount of time it takes for data to flow across the network. After that, signal strength is the magnitude of an electric field at a reference point positioned at a substantial distance from the transmitting antenna and this is stated in terms of the receiver's signal power or the voltage per length received by the reference antenna [11]. Lastly, Reliability is the delivery of genuine packets from source to destination in an acceptable period, which relies on the application type and architecture [12].

Throughput formula [13]:

Throughput = (frames per minute * frame carries) / 60

Latency formula [13]**:**

Latency = Transmission Time + Processing Delay + Propagation Time + Queuing Time

Signal Strength formula [14]:

RSSI=Pt-PL(d)

Pt = denoted the signal transmission power

PL(d) = denoted the route loss

Reliability formula [15]:

R(t) = 1 –F(t)

2.2 Critical Remarks of Existing Similar Application

2.2.1 Opensignal Application

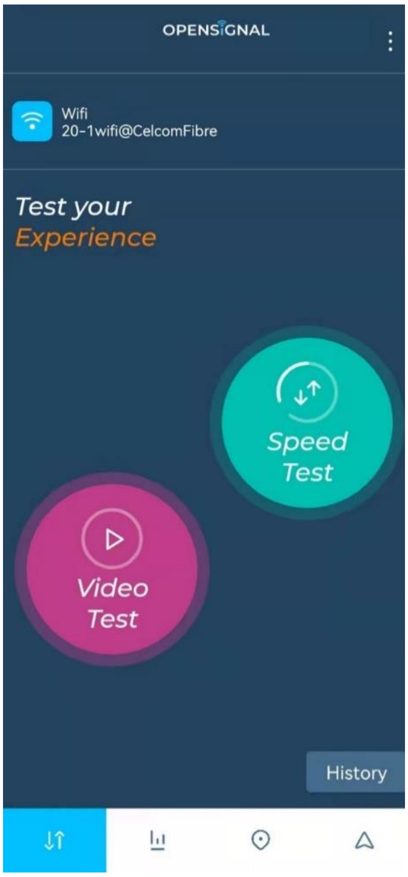
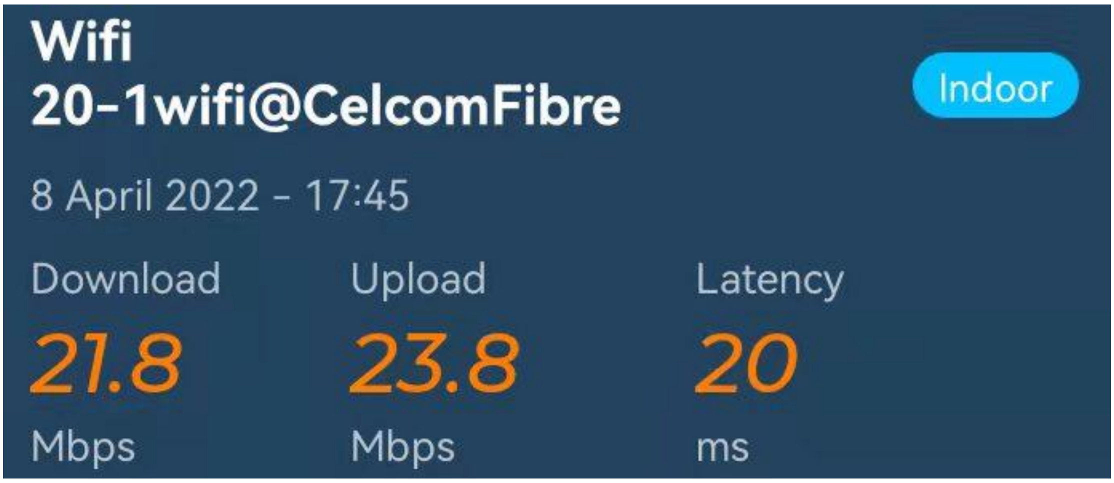


Figure 2.1 Opensignal Application

Opensignal collects network speed metrics from both user-initiated and automated testing. The bulk of measurements are obtained by automated tests (no user interaction), which are run separately and at random intervals to capture what users are experiencing at any given time. Several governmental agencies, including the FCC in the United States, have acknowledged this method as best practice. Opensignal does not have its own test servers. Opensignal measures the end-to-end consumer network experience, as well as the entire path from the user device to Content Delivery Networks (CDNs) like Google, Akamai, and Amazon. Under normal usage settings, Opensignal collects billions of individual measurements from over 100 million devices worldwide in every day, including readings in both indoor and outdoor areas. Because users spend the majority of their time indoors, the majority of our measurements will be obtained from indoor settings. We employ video content sources chosen to approximate normal user experience to create our video measure. Our metrics are structured in such a way that operators cannot adapt their networks to treat our traffic differently and so influence the findings without really improving their networks [16].



Wifi
20-1wifi@CelcomFibre

Indoor

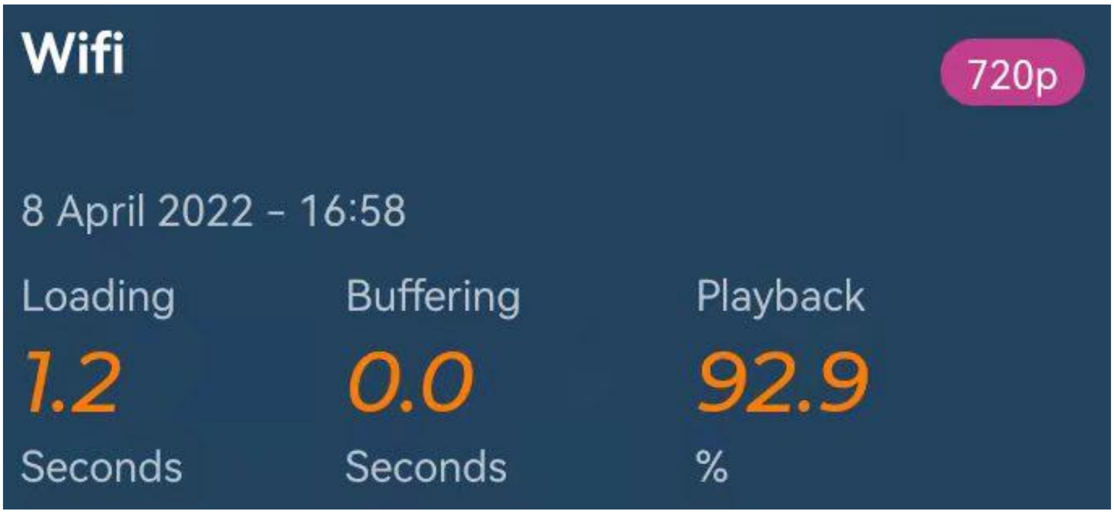
8 April 2022 - 17:45

Download 21.8 Mbps

Upload 23.8 Mbps

Latency 20 ms

Figure 2.2 Speed Test sample output by using Opensignal Application



Wifi

720p

8 April 2022 - 16:58

Loading 1.2 Seconds

Buffering 0.0 Seconds

Playback 92.9 %

Figure 2.3 Video Test sample output by using Opensignal Application

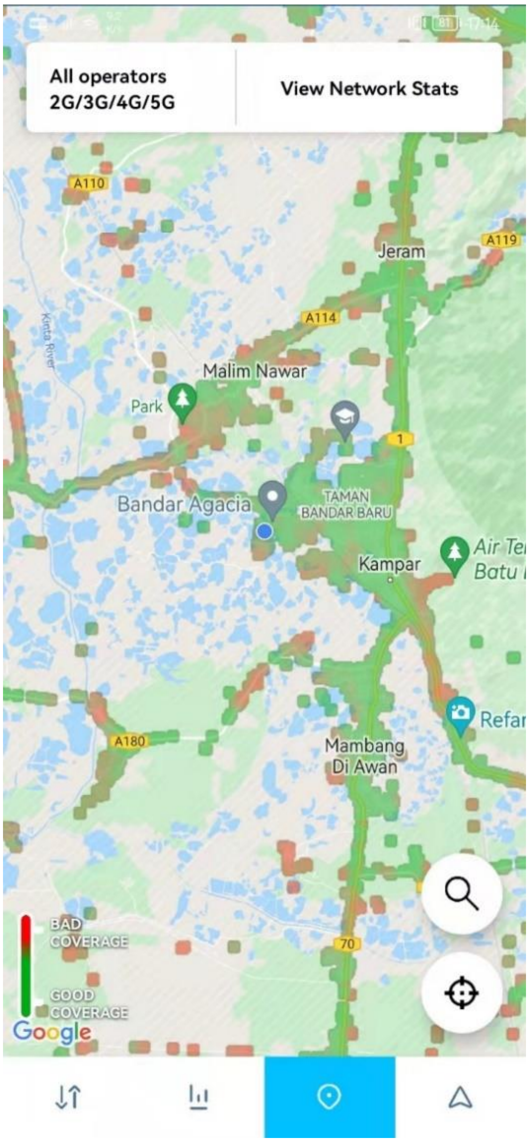


Figure 2.4 Network Status sample output in third page of Opensignal Application

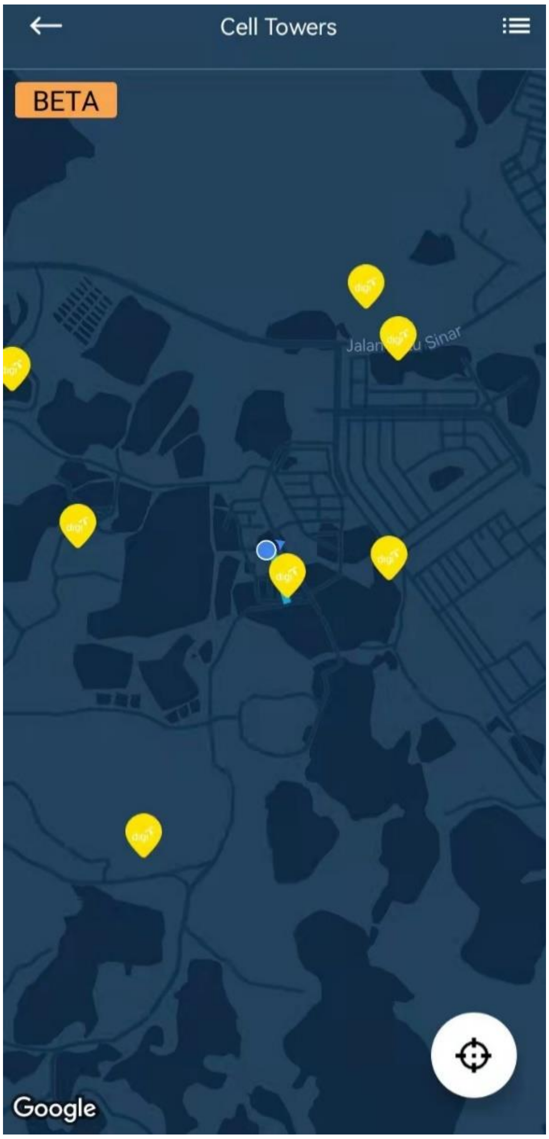


Figure 2.5 Cell Tower sample output fourth page of Opensignal Application

2.2.1.1 Opensignal Application Function:

1. Users have the ability to perform the network speed test.
2. Users have the ability to perform the video speed test.
3. Users have the ability to check the history of the network speed test and video speed test.
4. Users have the ability to view the network status according to the location.
5. Users have the ability to view the network availability.
6. Users have the ability to view the latency of the network.
7. Users have the ability to get the notification when the signal reception is going low

2.2.1.2 Opensignal Application Features:

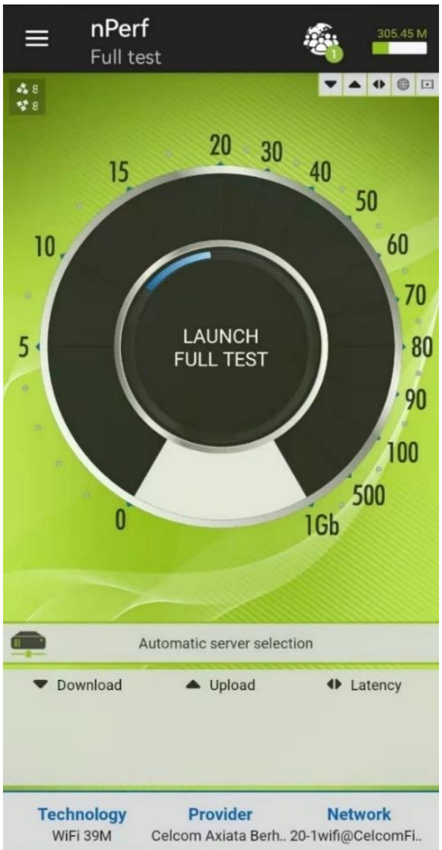
1. Enable users to know the throughputs after they click the speed test on the main page.
2. Enable users to know the video experience after they click the video test on the main page.
3. Provide a view of the type of current network used on the main page.
4. Enable users to know the network availability and the time can be selected by the user on the second page.
5. Enable users to know the network status such as bad or good network coverage by using the Google map function on the third page.
6. Enable users to know the location and quantity of cell towers by using the Google map function on the fourth page.
7. Provide the color-blind coverage maps in the setting.
8. Provide the notification when the user's signal goes below the level set by the user in the setting.
9. Provide the historical view of the speed experience and video experience.
10. Allow the user to accept or decline the data collection.

2.2.1.3 Opensignal Application Strength:

1. The application page design is pleasant and easy to get started.
2. Opensignal provides a function that can find the nearest cell tower by using the Google map function.

3. Opensignal will notify when the user’s signal reception goes below the level set by the user.
4. Opensignal will notify when the mobile data is finished.
5. Opensignal provides a theme for the people who are color blind.
6. Opensignal provides the view of network availability.
7. Opensignal provides the network coverage map by using the google map function and can select the operator and type of network selection.
8. Opensignal provides the historical view to the user, and the arrangement is pleasant.
9. Opensignal can identify the internet service providers and show them on the main page.
10. Opensignal provides two functions which are “Report a bug” and “Contact”, it is very friendly to the user when they are facing problems.

2.2.2 nPerf Application



Technology: WiFi 39M

Provider: Celcom Axiata Berh..

Network: 20-1wifi@CelcomFi..

Figure 2.6 nPerf Application

nPerf's goal is to assist users in determining the quality of their Internet connection. A bitrate test, no matter how strong it is, is insufficient to describe the quality of an Internet connection. Every year, nPerf collects millions of tests and billions of mobile network coverage measurements. The nPerf server network is comprised of over 1400 servers located across the world, with a total bandwidth of more than 5 Tb/s [17]. The speed test uses an innovative method to precisely assess your connection's latency, upload bitrate, and download bitrate. nPerf employs a worldwide dedicated server network that is optimized to supply enough bandwidth to overload your connection so that we can properly assess its bitrate. All broadband and mobile connections are supported by the nPerf speed test, including ADSL, VDSL, cable, optical fiber FTTH / FTTB, satellite, Wi-Fi, WiMAX, cellular 2G / 3G / 4G (LTE), and 5G. The nPerf system only gathers statistics that are geo-localized. There is no private information gathered. Data collected is used to characterize operator networks. All data collected becomes the property of nPerf SAS. The following information was gathered:

geolocation, network information, device type and model, system version, and test measures [18].

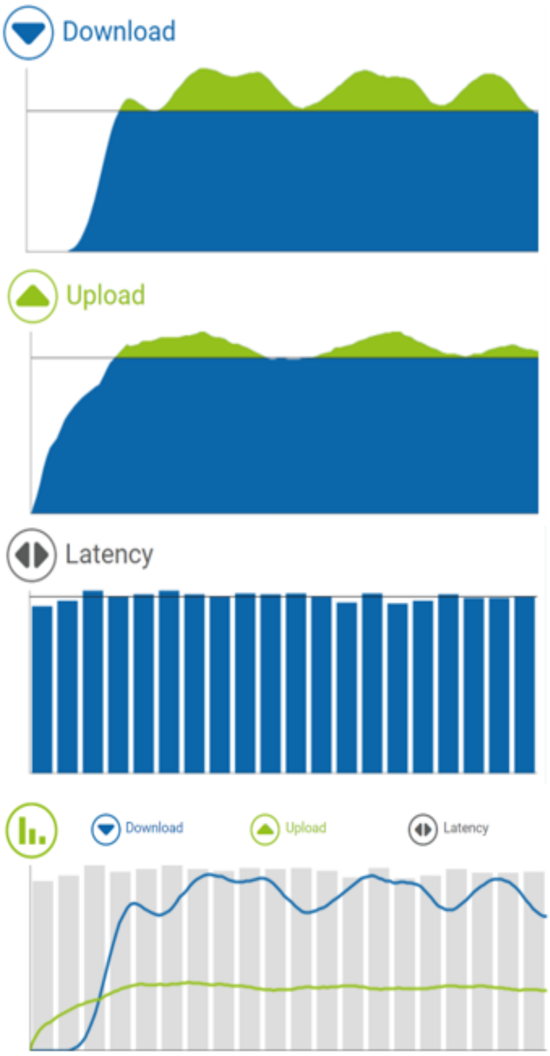


Figure 2.7: Sample output of nPerf Application

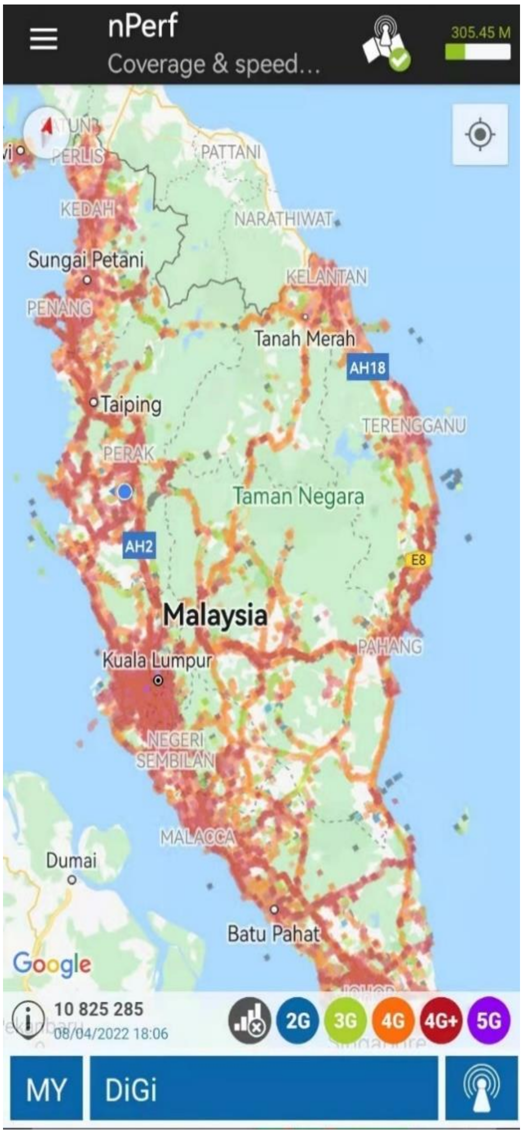


Figure 2.8: Coverage and speed maps of nPerf Application

2.2.2.1 nPerf Application Function:

1. Users have the ability to perform the full test.
2. Users have the ability to perform the speed test.
3. Users have the ability to perform the browsing test
4. Users have the ability to perform the streaming test
5. Users have the ability to check the history of the full test and speed test.
6. Users have the ability to view the coverage and speed maps.
7. Users have the ability to activate mapping.
8. Users have the ability to perform login and logout.

2.2.2.2 nPerf Application Features:

1. Enable users to know the whole network performance after they click the full test on the main page.
2. Enable users to know the throughputs after they click the speed test in the navigation bar.
3. Enable users to know the steaming experience after they click the streaming test in the navigation bar.
4. Enable users to know the browsing experience after they click the browsing test in the navigation bar.
5. Provide the historical view of the full test and speed test in the navigation bar.
6. Enable users to know the coverage and speed maps by using the Google map function in the navigation bar.
7. Provide users sign up, log in and log out to their account.
8. Enable users to active mapping by using the Google map function in the navigation bar.
9. Allow users to change the application's language to Chinese, English, Japanese and so on.

2.2.2.3 nPerf Application Strength:

1. The application page design is pleasant and easy to get started.
2. nPerf will show the jitter score after finishing the full test.
3. nPerf provide the language selection.
4. nPerf provide the network coverage map by using the google map function and can select the operator and type of network selection.

5. nPerf provide the historical view to the user, and the arrangement is pleasant
6. nPerf can identifier the internet service providers and show on the main page.
7. nPerf provide two function which is “Report a bug” and “Contact Us”, it is very friendly to the user when they are facing problem.
8. nPerf provide the active mapping for the user can have more accurate data.
9. nPerf provide the database for the user can sign up for an account and store the historical.

2.2.3 Speedtest Application (by Ookla)

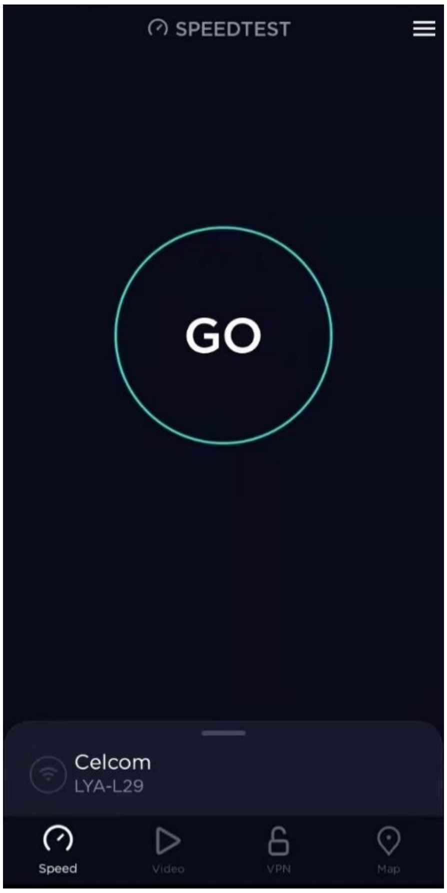
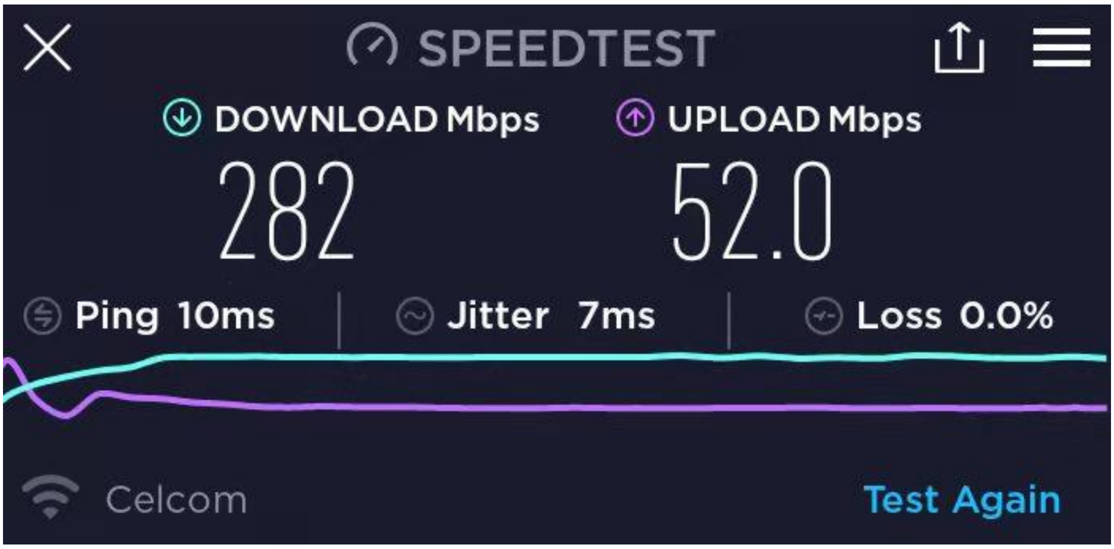


Figure 2.9: Speedtest Application (by Ookla)

Speedtest® by Ookla® is a system use to test the speed and performance of our internet connection. Every day, over ten million individual tests are actively initiated by users in areas and at times when their connectivity is critical to them. Since Speedtest's inception in 2006, an unprecedented total of more than 40 billion tests have been taken with Speedtest. Speedtest's mission to provide insight into network performance is what motivates them. Whether it's assisting our users in assessing the speed of their individual connections via the Speedtest application or publishing analysis on the state of global markets and related trends, the Speedtest team is always working to provide data that will help make the internet better and faster for everyone [19].Following that, each Speedtest test utilizes the Speedtest Server Network™, a worldwide network of high-performance servers ready to test the maximum sustained throughput of the user's connection (download and upload speeds) and report crucial network health data. Almost every country and major population centre on the earth has a Speedtest server. Learn more about the Speedtest Server Network and how Speedtest is the only tool that can analyse high-speed connections like 5G.



DOWNLOAD Mbps 282

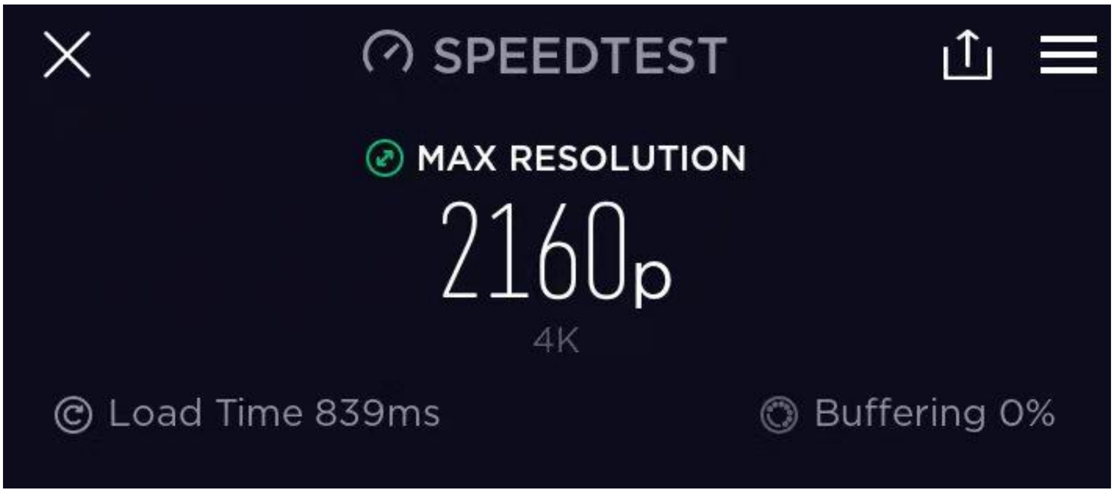
UPLOAD Mbps 52.0

Ping 10ms

Jitter 7ms

Loss 0.0%

Figure 2.10 Speed test sample output of Speedtest Application (by Ookla)



MAX RESOLUTION 2160p

4K

Load Time 839ms

Buffering 0%

Figure 2.11 Video test sample output of Speedtest Application (by Ookla)

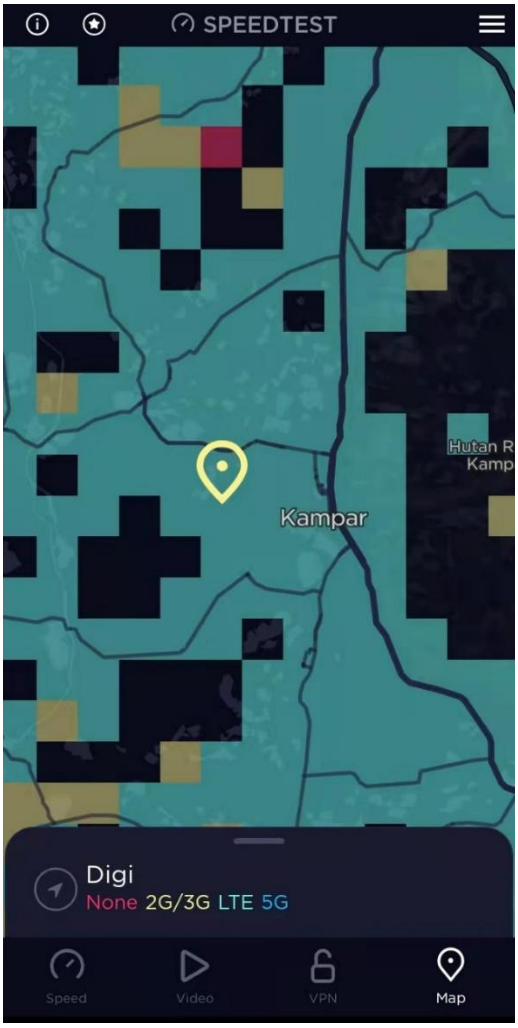


Figure 2.12 Coverage maps of Speedtest Application (by Ookla) in Kampar

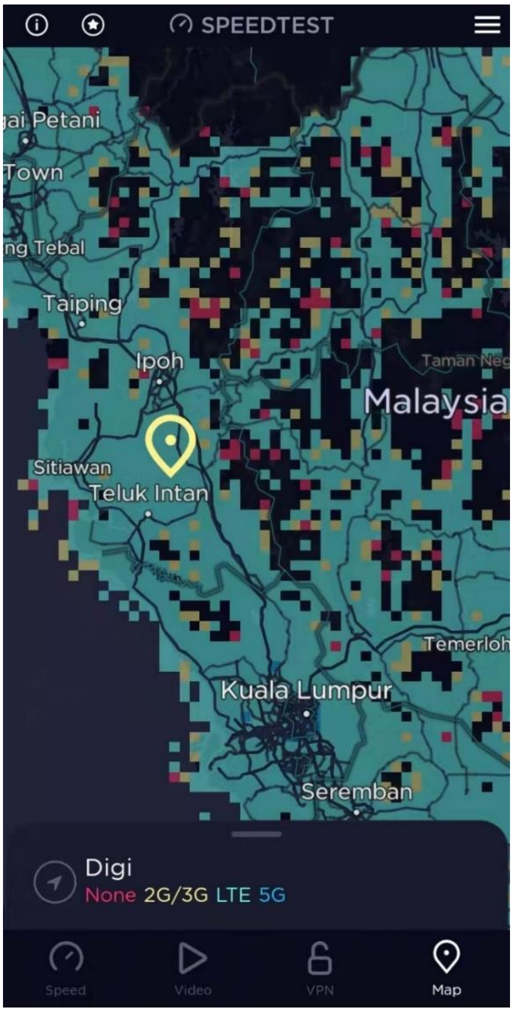


Figure 2.13 Coverage maps of Speedtest Application (by Ookla) in Malaysia

🟦 Blue = 5G

🟩 Green = 4G LTE

🟨 Yellow = 2G/3G

🟥 Red = None

2.2.3.1 Speedtest Application (by Ookla) Function:

1. Users have the ability to perform the speed test.
2. Users have the ability to perform the video test.
3. Users have the ability to perform the VPN mode.
4. Users have the ability to view the coverage map
5. Users have the ability to check the history of the speed test and video test.

2.2.3.2 Speedtest Application (by Ookla) Features:

1. Enable users to know the network performance after they click the speed test in main page.
2. Enable users to know the video experience after they click the video test in the second page.
3. Enable users to perform VPN mode in third page.
4. Enable users to know the coverage by using the Google map function in the fourth page.
5. Provide the historical view of the full test and speed test in navigation bar.
6. Allow users to select network signal scanning for improve the coverage map of the Speedtest Application in setting.
7. Allow users to give feedback to improve the Speedtest Application in the navigation bar.

2.2.3.3 Speedtest Application (by Ookla) Strength:

1. The application page design is pleasant and easy to get started.
2. Speedtest will show the jitter score after finishing the full test.
3. Speedtest provide the network coverage map by using the google map function and can select the operator and type of network selection.
4. Speedtest provide the historical view to the user, and the arrangement is pleasant.
5. Speedtest can identifier the internet service providers and show them on the main page.
6. Speedtest provides two functions which are “Speedtest Help” and “Give us Feedback”, it is very friendly to the users when they are facing problems.
7. Speedtest provides the packet loss percentage after finishing the speed test.
8. Speedtest provide the VPN mode.

2.3 Comparison of Similar Application

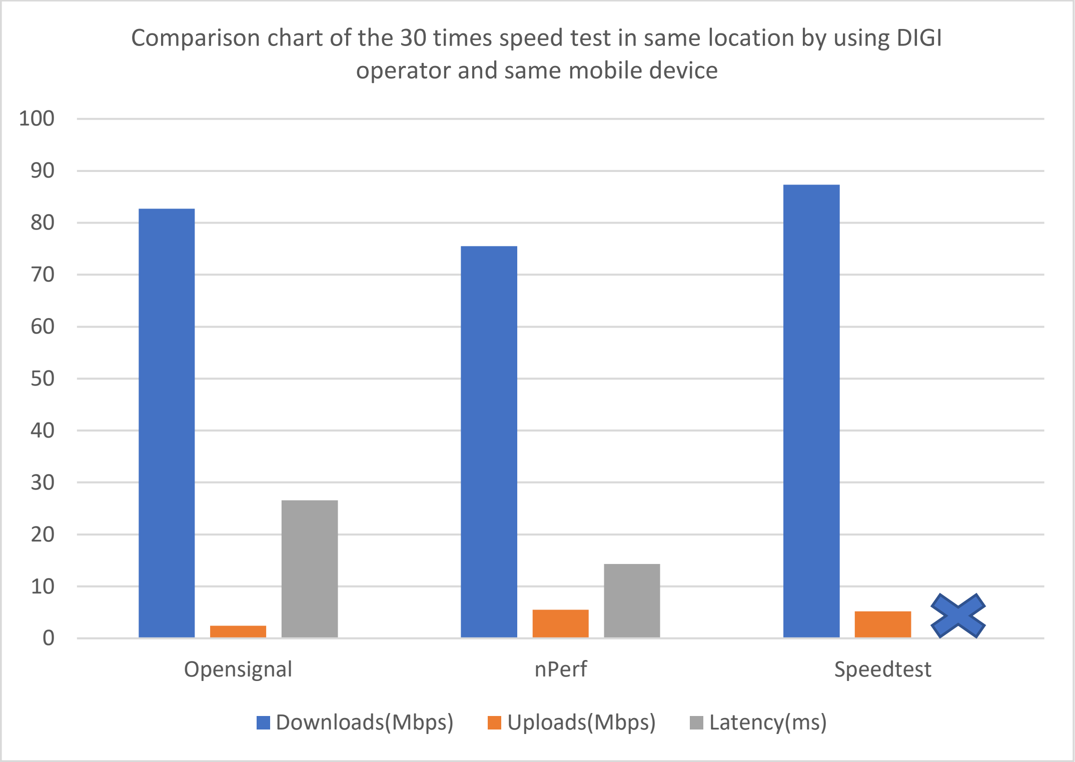


Figure 2.14 Comparison Chart

I have been performed 30 times of speed test in Opensignal, nPerf and Speedtest respectively at 12:23 noon. After that, we can observe that the downloads speed result are differences in these three applications, only the Speedtest do not have provide the latency result, although the speed test is performed in same location, using same device and operator, it will have differences result due to the formula differences and time differences, time will affect the no matter just differences a few minutes.

2.4 Limitation of Similar Application

2.4.1 Opensignal Application limitation:

1. Opensignal do not provide the telco recommendation based on location.
2. Opensignal do not have provide the browsing test.
3. Opensignal do not have provide the active mapping for the user to get more accurate data.
4. Speed test result do not have include the reliability scores.
5. Speed test result do not have include the jitter scores.
6. Opensignal do not have provide the VPN mode.
7. Opensignal do not have the function that can change the language of the application, default is English.

2.4.2 nPerf Application limitation:

1. nPerf do not provide the telco recommendation based on location.
2. nPerf do not have provide the view of network availability.
3. nPerf do not have provide a theme for the users who are color blind.
4. Speed test result do not have include the reliability scores.
5. nPerf do not have provide the VPN mode.
6. Opensignal do not have the function that can change the language of the application, default is English.
7. nPerf do not have provide function that can find the nearest cell tower by using Google map function.

2.4.3 Speedtest Application (by Ookla) limitation:

1. Speedtest do not provide the telco recommendation based on location.
2. Speedtest do not have provide the view of network availability
3. Speedtest do not have provide a theme for the users who are color blind.
4. Speed test result do not have provide the reliability scores.
5. Speed test result do not have provide the latency scores.
6. Speedtest do not have the function that can change the language of the application, default is English.
7. Speedtest do not have provide function that can find the nearest cell tower by using Google map function.

2.5 Solution for the limitations

Table 2.1 Solution for limitation

Name of Application	Solution for the limitation
Opensignal	1. Develop telco recommendation system by using the data provide by the user after they finish the speed test based on their location. 2. Develop a browsing test for the user can test their browsing performance. 3. Develop a mapping function for the user to have more accurate data. 4. Develop the reliability scores in the speed test for the user can more understand their network performance. 5. Develop the jitter scores in the speed test for the user can more understand their network stability. 6. Develop the VPN function for the user can have a more secure network. 7. Develop multiple languages function for the user who are do not understand English.
nPerf	1. Develop telco recommendation system by using the data provide by the user after they finish the speed test based on their location. 2. Develop a function for the user can view their network availability. 3. Develop a theme for the users who are color blind can have a better experience. 4. Develop the reliability scores in the speed test for the user can more understand their network performance.

	5. Develop the VPN function for the user can have a more secure network. 6. Develop multiple languages function for the user who are do not understand English. 7. Develop a function that can find the nearest cell tower by using Google map function.
Speedtest Application (By Ookla)	1. Develop telco recommendation system by using the data provide by the user after they finish the speed test based on their location. 2. Develop a function for the user can view their network availability. 3. Develop a theme for the users who are color blind can have a better experience. 4. Develop the reliability scores in the speed test for the user can more understand their network performance. 5. Develop the latency scores in the speed test for the user can more understand their network stability. 6. Develop the VPN function for the user can have a more secure network. 7. Develop multiple languages function for the user who are do not understand English.

2.6 Summary

Table 2.2 Proposed Project Table

Functions	Opensignal	nPerf	Speedtest	Proposed Project
Telco recommendation based on location	✗	✗	✗	✓
Provide identifier the Internet Service Providers	✓	✓	✓	✓
Provide speed test	✓	✓	✓	✓
Get notification when bad signal reception	✓	✗	✗	✓
Provide historical view	✓	✓	✓	✓
Provide "Report bug" and "FAQ"	✓	✓	✓	✓
Provide network coverage map	✓	✓	✓	✓
Provide jitter scores in speed test result	✗	✗	✓	✓
Provide reliability scores in speed test result	✗	✗	✗	✓
Provide latency scores	✓	✓	✗	✓
Provide multiple language selection	✗	✓	✗	✓

To sum up, Opensignal has the most function as mentioned above. However, it still lacks some importance function, such as do not have provide the telco recommendation, jitter scores in speed test result, reliability scores in speed test result and do not have the multiple language selection. But it also provides two unique function such as view of nearest cell tower by using the Google map function and get notification when bad signal reception. In my proposed project, I will propose a significate function, telco recommendation by using opensource database, it can let users easily to identifier the better telco based on their location.

CHAPTER 3 PROPOSED METHOD/APPROACH

3.1 Methodology

3.1.1 Comparison of different methodology

3.1.1.1 Scrum

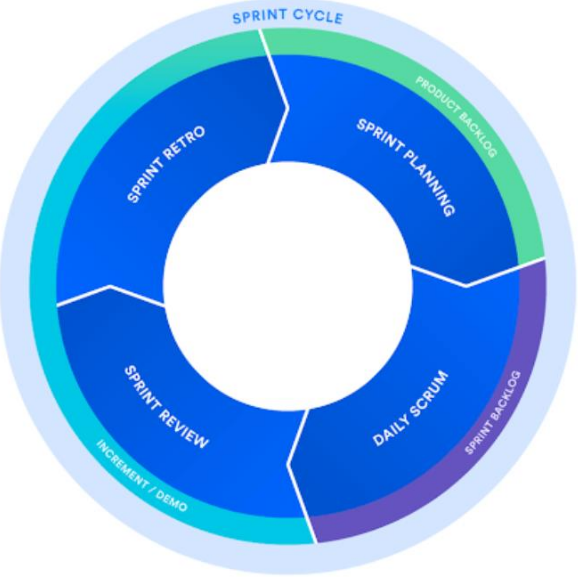


Figure 3.1 Scrum

Scrum is an agile software development methodology that prioritizes cooperation, teamwork, and the capacity to react fast to changing needs. It is a commonly utilized approach in the software industry for managing and finishing large projects. In Scrum, a project is broken up into brief sprints, which are generally two to four weeks long. The team works to finish a number of user stories or prioritized activities that are taken from the project backlog during each sprint.

The Product Owner, the Scrum Master, and the Development Team are the three positions that make up the Scrum framework. The Product Owner is in charge of controlling the project backlog's priorities and making sure the team is focusing on the most crucial and worthwhile tasks. The Scrum Master is in charge of enabling the Scrum process, removing obstacles that get in the way of the team's productive work, and making sure the team is adhering to Scrum best practices. The Development Team is in charge of providing a potentially shippable product increment at the conclusion of each sprint, as well as fulfilling the tasks listed in the sprint backlog.

Scrum also involves a number of rituals or gatherings to encourage teamwork and communication. These include the Daily Scrum, a quick daily stand-up meeting where team members share progress, talk about obstacles, and plan the following day, the Sprint Review, where the team presents the work completed during the sprint and receives feedback from stakeholders, and the Sprint Retrospective, where the team evaluates the sprint and pinpoints areas for improvement.

In conclusion, Scrum is, all things considered, a highly collaborative and adaptive framework that enables teams to efficiently collaborate and produce high-quality software products in a timely and effective way.

3.1.1.2 Feature-Driven Development

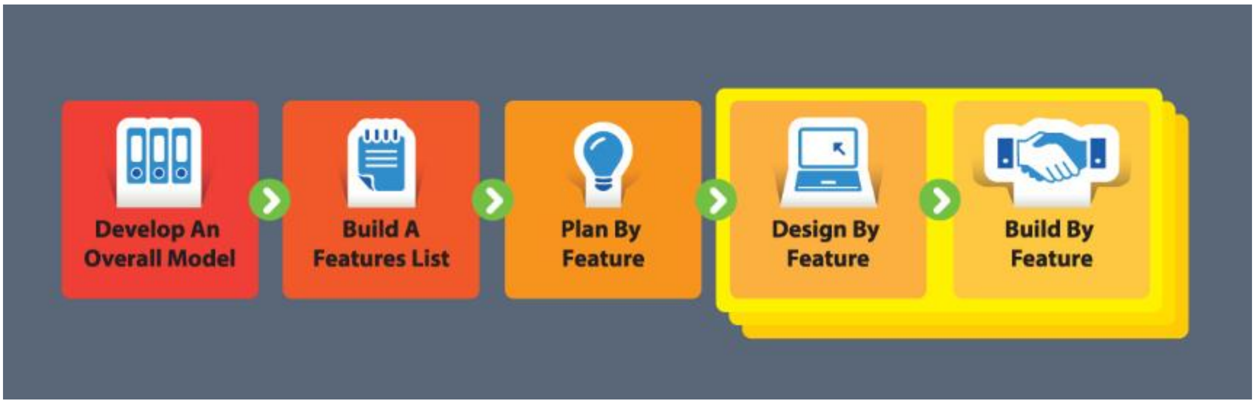


Figure 3.2 Feature-Driven Development

A software development process that emphasizes timely and effective delivery of functional software features is called "feature-driven development" (FDD). Jeff De Luca and Peter Coad created FDD in the late 1990s. It is based on their experience creating huge software systems. FDD is built on a set of five fundamental tasks: creating a general model, creating a list of features, planning by features, designing by features, and constructing by features. These tasks are completed iteratively, with each iteration concentrating on delivering a set of functional characteristics.

A high-level perspective of the software system serves as the overarching model in FDD, which directs the development process. The development team and stakeholders' feedback are used to generate the feature list, which is a prioritized list of items that must be implemented. Planning per feature entails dividing the feature list into smaller, more manageable chunks and creating a thorough strategy for putting each feature into practice. When you design by feature, you create a thorough design for each feature that will later be utilized to direct the implementation process.

Building by feature entails putting each feature into practise one at a time before incorporating it into the whole system. In order to make sure the system is operating properly, FDD emphasises the value of testing throughout the development process and encourages teams to employ automated testing methods.

Overall, FDD is a simple and adaptable technique that emphasises the value of regularly providing functional software features. It works effectively for projects with clear criteria and a desire for quick delivery of usable software.

3.1.1.3 Lean software development

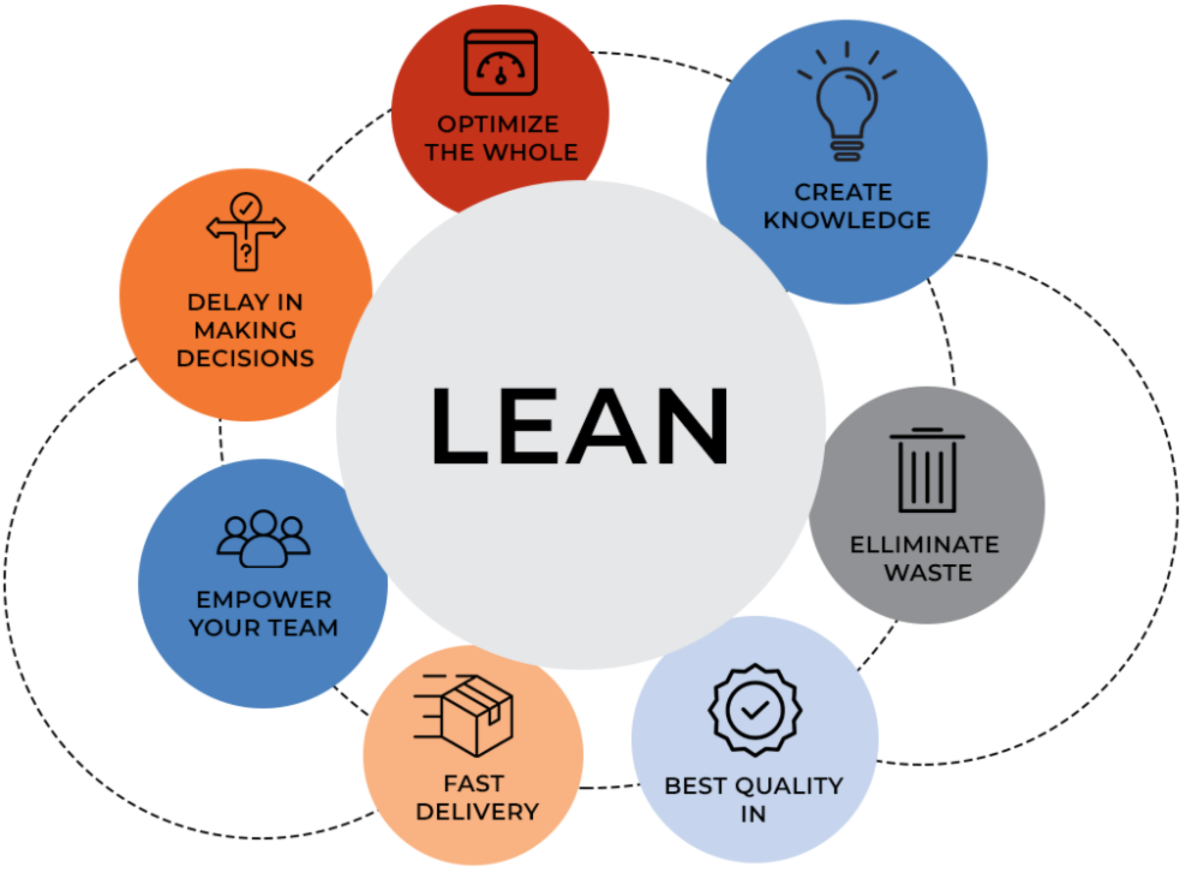


Figure 3.3 Lean software development

A process known as lean software development emphasizes maximizing customer value while reducing waste. It is founded on the lean manufacturing tenets that the Toyota Production System pioneered. Eliminating waste, enhancing learning, empowering the team, delivering as quickly as feasible, and incorporating integrity into the system are some of the fundamental tenets of lean software development.

Finding and removing activities that don't advance the software development process' goals is the elimination of waste. Wasteful practices include delay, overproduction, and rework. Teams should continually learn from their experiences in order to enhance their processes and deliverables. This is known as amplifying learning. Giving team members the power to decide and take responsibility for their job while putting an emphasis on teamwork and shared responsibility constitutes empowering the team.

Teams should strive to produce functioning software as rapidly as feasible while retaining quality in order to release as quickly as possible. Finally, teams should concentrate on creating high-quality software that satisfies customer demands and is simple to maintain over time in order to instill integrity into the system. Teams working on projects with shifting needs, where efficiency, adaptability, and quality are crucial, benefit from lean software development.

3.1.2 Chosen a Methodology

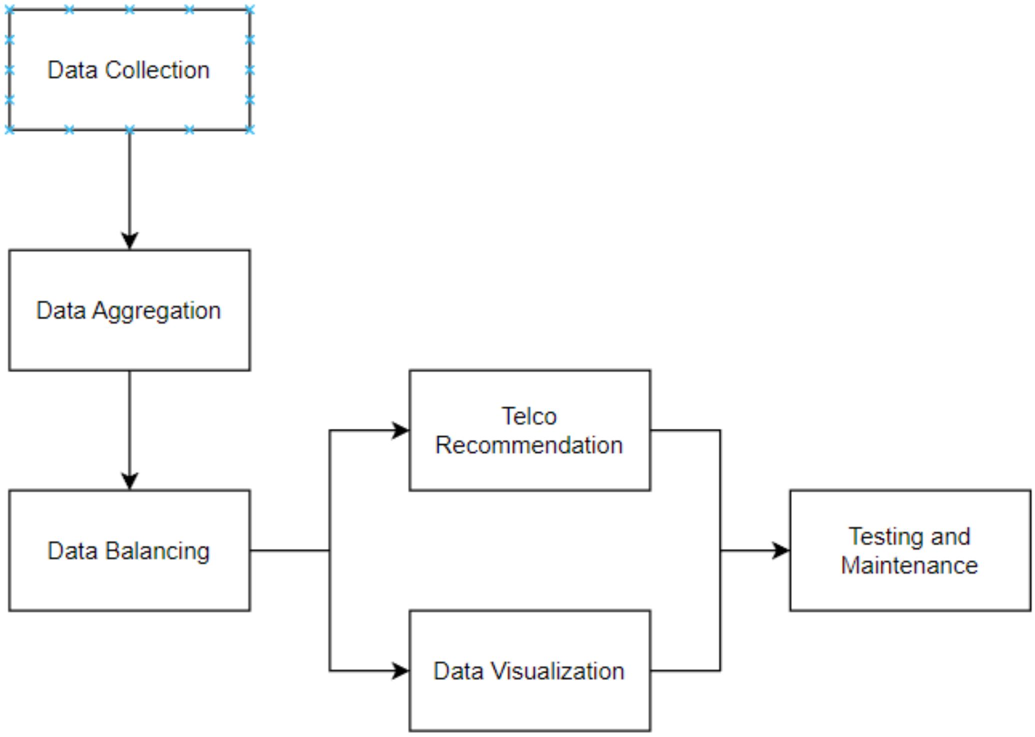


Figure 3.4 Methodology

Scrum is well-suited for projects with rapidly changing requirements, and this project involves multiple components, including data collection, aggregation, balancing, recommendation, visualization, and testing. With Scrum, it can work in iterations and prioritize the most important features or components for each iteration and allow to deliver value to the stakeholders incrementally. Additionally, Scrum emphasizes continuous feedback from stakeholders, which is important for a project that involves data analysis and user recommendations.

This project aims to provide telco recommendations based on user location using speed test data. The six components involved in this project are Data Collection, Data Aggregation, Data Balancing, Telco Recommendation, Data Visualization, and Testing and Maintenance. Data Collection involves gathering user location data from various sources, such as GPS sensors and mobile networks. Data Aggregation combines the collected data into a single dataset, which is then processed by Data Balancing to ensure fairness and accuracy. Telco Recommendation will use Haversine Algorithm to recommend the most suitable telco plan for the user based on their location and average score of download speed and upload speed. Data Visualization provides a graphical representation of the recommendation results, making it easier for the user to understand and compare. Finally, Testing and Maintenance ensures that the system operates correctly, by testing and debugging the system, and maintaining it over time. By following these six components, this project can provide a useful and reliable telco recommendation service for users based on their location, which can help them save money and improve their mobile network experience.

3.2 System Requirement

3.2.1 Hardware

3.2.1.1 Laptop

The hardware involved in this project is Laptop. The computer issued for the process develop the application for the telco recommendation using optimised OpenSignal data system. The software involve in this project is Flask, Shimadzu, Hybrid Apps.

Table 3.1 Specifications of laptop

Description	Specification

Model	Victus by HP Laptop 16-d0161TX
Processor	AMD Ryzen™ 7 5800H
Operating System	Windows 11
Graphic	NVIDIA GeForce RTX3050ti 4GB DDR4
Memory	16GB DDR4 RAM
Storage	1.5TB NVMe SSD

3.2.1.2 Vivo X80

Table 3.2 Specifications of Vivo X80

Description	Specification
Model	Vivo X80 / V2144
Processor	Dimensity 9000 Octa-core
Operating System	Android
Memory	12GB RAM+4GB Virtual RAM
Storage	256GB
LTE	√

3.2.1.3 Samsung Galaxy S6

Table 3.3 Specifications of Samsung Galaxy S6

Description	Specification
Model	Samsung Galaxy S6
Processor	Exynos 7420 Octa
Operating System	Android
Memory	3GB RAM
Storage	32GB
LTE	√

3.2.1.4 Oppo F11 Pro

Table 3.4 Specifications of Oppo F11 Pro

Description	Specification
Model	Oppo F11 Pro
Processor	Mediatek MT6771 Helio P70
Operating System	Android
Memory	6GB RAM
Storage	128GB
LTE	√

3.2.1.5 Xiaomi Redmi Note 8 Pro

Table 3.5 Specifications of Xiaomi Redmi Note 8 Pro

Description	Specification
Model	Samsung Galaxy S6
Processor	Mediatek Helio G90T
Operating System	Android
Memory	4GB RAM
Storage	128GB
LTE	√

3.2.1.5 Xiaomi Mi 8 Pro

Table 3.6 Specifications of Xiaomi Mi 8 Pro

Description	Specification
Model	Xiaomi Mi 8 Pro
Processor	Qualcomm SDM845 Snapdragon 845
Operating System	Android
Memory	6GB RAM
Storage	128GB
LTE	√

Above are the phone that I used for collecting the network speed test data.

3.2.2 Software

3.2.2.1 Apache2

Figure 3.5 Apache2

On Linux operating systems, including Ubuntu, Apache2 is a well-liked web server programmed. Both static and dynamic web sites can be served by this open-source program. When a client, such as a web browser, makes a request, Apache2 listens for it and responds by delivering web pages or other resources. The server-side programming languages PHP and Perl, SSL encryption, virtual hosting, and other capabilities are all supported by Apache2, which is highly flexible and supports a wide range of features and modules. It is also popular among many web hosting companies since it is effective and lightweight.

3.2.2.2 Google My Maps

Figure 3.6 Google My Maps

Google My Maps offers the user access to Google Maps and a lot of the information you'll get when you use Google Maps. It functions similarly to Google's other cloud-based authoring tools in many ways. You may create maps on your own or share them with other people to collaborate on projects. It even integrates with Google Docs, allowing you to arrange your maps alongside other papers. You may also import information from Google Docs and Spreadsheets into your personalized map. Google

My Maps have two features that will be used in this project. The first feature is collaboration. The user may share their own map with particular persons, anybody with the link, or people in the user's Google organisation, just as they would any other Google Drive file. The second element is the draw or plot functionality, which consists mostly of the draw the line or plot a marker tool. Users can draw a line without adding markers, but it will not signify anything unless it is contextualised.

3.2.2.3 Jupyter Notebook

Figure 3.7 Jupyter Notebook

The Jupyter Notebook is an open-source web tool for creating and sharing documents with live code, equations, visualizations, and text. People at Project Jupyter manage Jupyter Notebook. Jupyter Notebooks are a fork of the IPython project, which used to have its own IPython Notebook project. Jupyter can support some programming languages such as Python and Julia. Jupyter comes with the IPython kernel, which allows you to develop Python programs, although there are presently over 100 alternative kernels available. So that, this project will use Jupyter Notebook to do the data pre-processing for the dataset due to the dataset will have some data may contain NaN value, I will use Jupyter Notebook to fill up all the NaN value to have an accurate dataset.

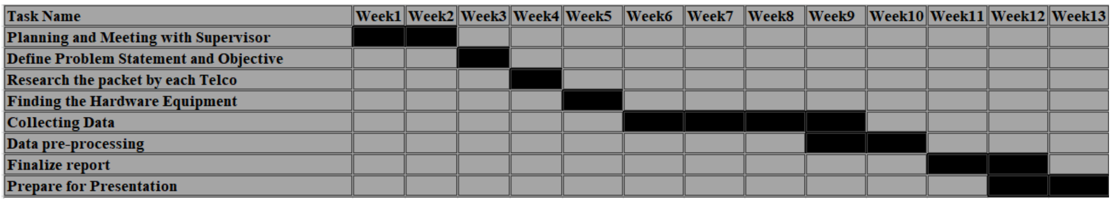
3.2.2.4 GitHub

Figure 3.8 GitHub

GitHub is a service that hosts Git repositories, but it also adds many of its own services. Git is a command-line program, whereas GitHub is a Web-based graphical interface. It also includes access control and a variety of collaborative capabilities, such as wikis and basic task management tools, for each project. Therefore, GitHub will be use in this project to store the speed test data in Kampar area and set as public, let others view and utilize the dataset to lead the world become more connected via cellular networks because the data accuracy is crucial, especially for the bandwidth and other metric of the network performance.

3.3 Timeline

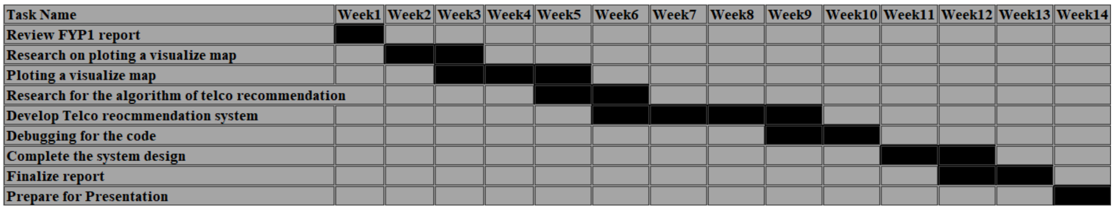
3.3.1 FYP1



Task Name	Week1	Week2	Week3	Week4	Week5	Week6	Week7	Week8	Week9	Week10	Week11	Week12	Week13
Planning and Meeting with Supervisor	■	■											
Define Problem Statement and Objective			■										
Research the packet by each Telco				■									
Finding the Hardware Equipment					■								
Collecting Data						■	■	■	■				
Data pre-processing									■	■			
Finalize report											■	■	
Prepare for Presentation												■	■

Figure 3.9 Timeline for FYP I

3.3.2 FYP2



Task Name	Week1	Week2	Week3	Week4	Week5	Week6	Week7	Week8	Week9	Week10	Week11	Week12	Week13	Week14
Review FYP1 report	■													
Research on ploting a visualize map		■	■											
Ploting a visualize map			■	■	■									
Research for the algorithm of telco recommendation					■	■								
Develop Telco reocmmendation system						■	■	■	■					
Debugging for the code									■	■				
Complete the system design											■	■		
Finalize report												■	■	
Prepare for Presentation														■

Figure 3.10 Timeline for FYP II

3.4 Estimated Cost

Table 3.7 Estimated Cost Table

No.	Code	Description	Qty(unit)	Price(RM)	Discound	Tax	Amount	DATE
1	PRE-CFLEXI	POSTPAID UPFRONT PAYMENT (XPAX POSTPAID 40) 013-8508014	1	40	0	0	40	15/07/2022
2	PRE-HLEXI	POSTPAID UPFRONT PAYMENT (HOTLINK POSTPAID 40) 017-3640193	1	40	0	0	40	15/07/2022
3	PRE-UMFLEXI	POSTPAID UPFRONT PAYMENT (HERO P38) 011-62441553	1	38	0	0	38	16/07/2022
4	PRE-TFLEXI	POSTPAID UPFRONT PAYMENT (YES INFINITE BASIC) 018-7012765	1	61.5	0	0	61.5	16/07/2022
5	PRE-YFLEXI	YES POSTPAID PAYMENT (018-7012765)	1	10.35	0	0	10.35	7/8/2022
6	OTHER-SRV	POSTPAID TERMINATION FEE FOR YES 018-7012765	1	10	0	0	10	7/8/2022
7	OTHER-SRV	POSTPAID TERMINATION FEE FOR UMOBILE 011-62441553	1	10	0	0	10	7/8/2022
8	OTHER-SRV	POSTPAID TERMINATION FEE FOR CELCOM 013-8508014	1	10	0	0	10	7/8/2022

3.5 Concluding Remark

In conclusion, the development of a successful project requires a rigorous and meticulous approach to system design and implementation. The system must meet all the requirements outlined in the project brief, including both hardware and software components. The hardware must be carefully selected to ensure optimal performance and compatibility with the software components, and the software must be thoroughly tested and optimized to achieve maximum efficiency and functionality. Furthermore, the timeline for the project must be carefully planned and followed to ensure that all project milestones are met on time. The success of the project will depend on the team's ability to collaborate effectively and communicate clearly throughout both FYP1 and FYP2. Regular team meetings and progress reports will help to ensure that everyone is on track and that any potential issues are identified and addressed promptly. It is also essential to estimate the costs of the project accurately and to adhere to the budget throughout the project's lifespan. Any unforeseen expenses must be carefully considered and approved by the project supervisor before any additional costs are incurred. Ultimately, the success of this project will depend on the team's dedication and hard work. By following a systematic approach to system design and implementation, adhering to the timeline, and managing costs effectively, the team will be well-positioned to achieve the desired outcomes of the project. The skills and experience gained from this project will undoubtedly prove invaluable in the team members' future academic and professional pursuits.

CHAPTER 4

4.1 System Design

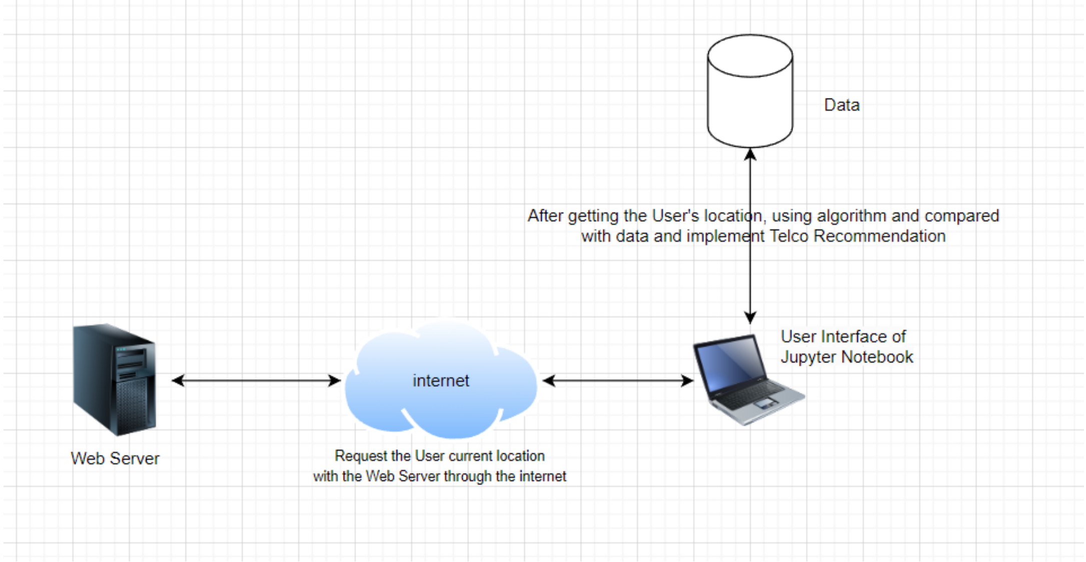

Data
After getting the User's location, using algorithm and compared with data and implement Telco Recommendation
User Interface of Jupyter Notebook
internet
Web Server
Request the User current location with the Web Server through the internet

Figure 4.1 System overview

This is the overview of the system design for a Telco recommendation system based on user location. The system is designed to recommend Telco providers to users based on their current location and the maximum distance they are willing to travel. The system is divided into four components: Jupyter notebook, Data, Web Server, and Internet. This report will discuss each component in detail and how they work together to create the overall system.

4.1.1 Jupyter Notebook Component

The Jupyter Notebook is a web application that allows users to create and share documents that contain live code, equations, visualizations, and narrative text. In this system, the Jupyter Notebook component is used to develop and test the code for the recommendation system. The code in the Jupyter Notebook retrieves the user's location, calculates the distance between the user and Telco providers, and recommends the top Telco providers within the user's preferred distance. The code is written in Python and utilizes libraries such as Selenium and Pandas.

4.1.2 Data Component

The Data component contains the Telco provider data that the recommendation system uses to make recommendations to the user. The data is stored in an Excel file that contains information such as the Telco provider name, location, download speed, upload speed, and latency. The data is loaded into the Jupyter Notebook component using the Pandas library. The data is then processed and filtered based on the user's location and preferred distance.

4.1.3 Web Server Component

The Web Server component is responsible for hosting the Telco recommendation system and making it available to users. The system is deployed on a web server that is accessible through the internet. The user interacts with the system through a web interface that allows them to input their current location and preferred distance. The web interface sends the user's input to the Jupyter Notebook component, which processes the data and returns the top Telco recommendations to the user.

4.1.4 Internet Component

The Internet component provides the network connectivity that allows the Telco recommendation system to function. The system relies on the internet to retrieve the user's location, access the Telco provider data, and communicate with the user through the web interface. The system also utilizes the internet to connect to the Selenium WebDriver, which is used to automate web browser interactions and retrieve the user's location.

4.1.5 Conclusion

In conclusion, the Telco recommendation system is designed to recommend Telco providers to users based on their current location and the maximum distance they are willing to travel. The system is divided into four components: Jupyter Notebook, Data, Web Server, and Internet. The Jupyter Notebook component retrieves the user's location, calculates the distance between the user and Telco providers, and recommends the top Telco providers within the user's preferred distance. The Data component contains the Telco provider data, and the Web Server component hosts the system and provides a web interface for the user. Finally, the Internet component provides the network connectivity that allows the system to function. Together, these components create a Telco recommendation system that is accurate, efficient, and user-friendly.

4.2 Overall system Flow Chart

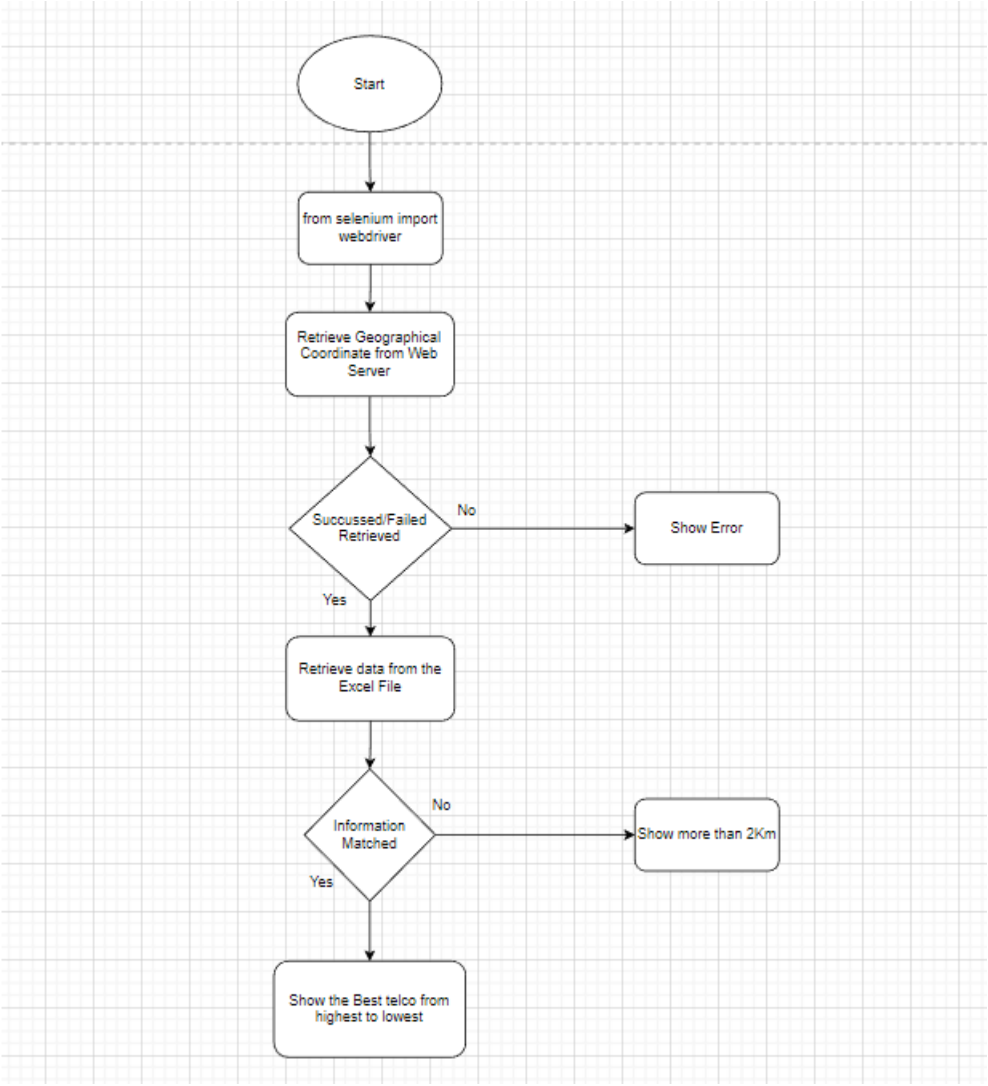

Start
from selenium import webdriver
Retrieve Geographical Coordinate from Web Server
Succussed/Failed Retrieved
No
Show Error
Yes
Retrieve data from the Excel File
Information Matched
No
Show more than 2Km
Yes
Show the Best telco from highest to lowest

Figure 4.2 Flow Chart

Once the system started, the first thing to do is using python to perform from selenium import webdriver. Furthermore, connect with the web driver to retrieve the geographical coordinate of the user and stored it in the Xpath and using python to retrieve the geographical coordinate from the Xpath. After that, the geographical coordinated will be stored into a variable and print the variable on screen. If failed, then error occurs. If success, it will proceed to retrieve the data from the Excel file and start to use haversine algorithm to calculate the distance between the geographical coordinate of data and user's geographical coordinate. If within 2km, it will provide the data that sorted using python, and recommend the best telco that have highest download speed plus highest upload speed from highest to lowest. If more than, it will show “No Telco provider found within 2 km.”

4.3 Functional Requirement

4.3.1 Telco recommendation System

4.3.1.1 Uses Selenium WebDriver to automate the process of finding the current location of the user:

The system is able to use Selenium WebDriver to automate the task of finding the current location of the user. Selenium WebDriver is a popular tool used for automating web browser interactions. The task involves launching the Microsoft Edge browser and navigating to a specific website called "https://findmylocation.org/". This website is designed to provide location-based services by using the location services feature of the user's device. Once the website is launched, the script waits for the longitude element to load, which is a specific element on the web page that contains the longitude and latitude values.

4.3.1.2 Telco Recommendation:

This system must have the ability to help the users who are looking to select a new telecommunications provider or switch to a different one. The system's ability to recommend a Telco provider based on the user's location and the provider's download and upload speeds, and latency can help users make more informed decisions about which provider to choose. And the system's use of the haversine algorithm to calculate the distance between the user's location and the Telco provider's location is critical in ensuring that the user is provided with accurate and reliable recommendations. This algorithm considers the curvature of the Earth's surface, which can be important when calculating distances over large areas.

4.3.1.3 Visualize the data in Google my map:

This system must have the ability to create a map that shows the speed test results for different locations based on the Longitude and Latitude inside the Excel file. It reads in the speed test data from an Excel file and then creates a new Google Map. Each location is marked on the map with a colored marker, with the color of the marker indicating the speed test result for that location. The map also includes a color scale that shows the range of speeds that correspond to different colors.

4.3.2 Speed Test System

4.3.2.1 Download Speed Test:

This system must have the ability to allows user to measure how fast user's internet connection is at downloading files. This is important because it affects how quickly user can load web pages, stream videos, and download files.

4.3.2.2 Upload Speed Test:

This system must have the ability to measure how fast user's internet connection is at uploading files. This is important because it affects how quickly user can send files, such as email attachments or uploading files to a cloud storage service.

4.3.2.3 Ping and Jitter Test:

This system must have the ability to allows user to measure the quality of user's internet connection. Ping refers to the time it takes for a signal to travel from user's device to a server and back, while jitter measures the variability in the delay of those signals. This is important because it affects how quickly user's device can communicate with other devices on the internet.

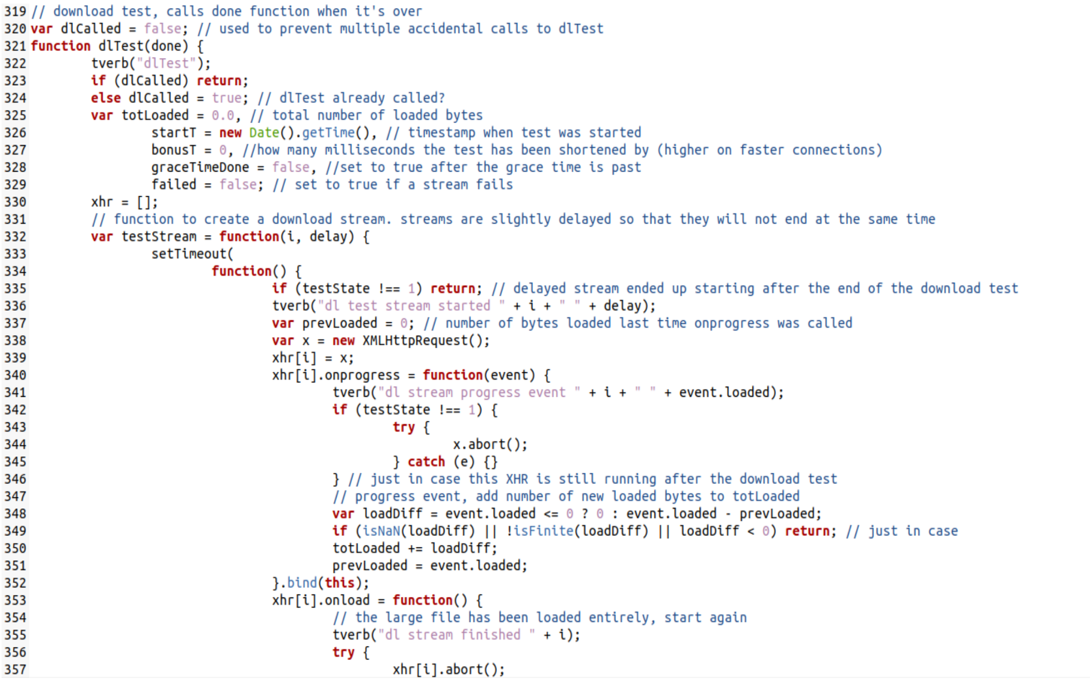
4.3.2.4 Retrieve Data from database managed by PhpMyAdmin and show in table:

This system must have the ability to retrieving data from a database managed by PhpMyAdmin and displaying it in a table format. PhpMyAdmin is a popular web-based application used to manage MySQL databases. The task involves accessing the database using PhpMyAdmin and executing a SQL query to retrieve the desired data. The retrieved data can then be formatted and displayed in a table on a web page using HTML and CSS. The table may contain columns and rows that represent different fields and records of the data. The displayed data can be sorted and filtered based on specific criteria to make it easier for users to find and analyze the information they need.

4.4 Algorithm Design

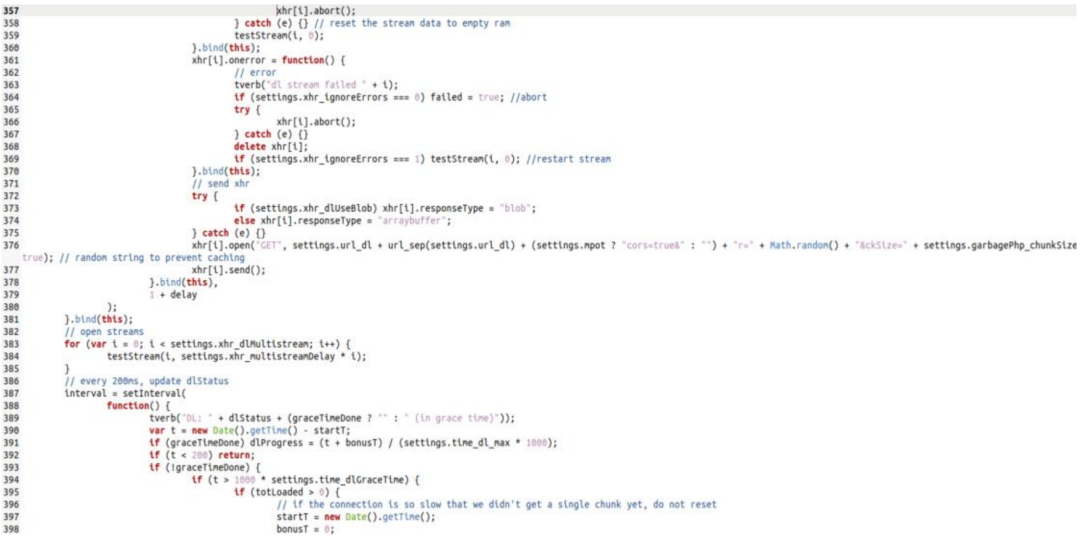
4.4.1 Download Test Algorithm

The download test algorithm measures the speed of data transfer from the server to the client. It works by requesting a file of a predetermined size from the server and measuring the time it takes to download the file. The download speed is then calculated by dividing the size of the file by the time taken to download it.



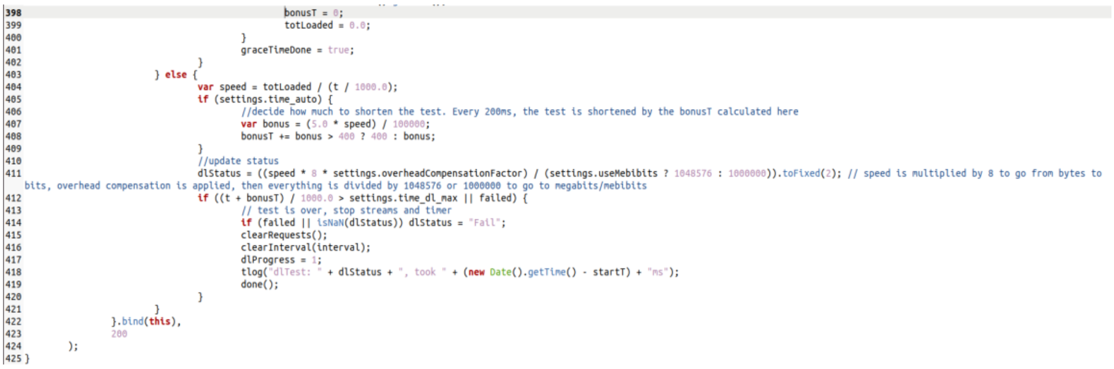
```
// download test, calls done function when it's over
var dlCalled = false; // used to prevent multiple accidental calls to dlTest
function dlTest(done) {
    tverb("dlTest");
    if (dlCalled) return;
    else dlCalled = true; // dlTest already called?
    var totLoaded = 0.0, // total number of loaded bytes
        startT = new Date().getTime(), // timestamp when test was started
        bonusT = 0, //how many milliseconds the test has been shortened by (higher on faster connections)
        graceTimeDone = false, //set to true after the grace time is past
        failed = false; // set to true if a stream fails
    xhr = [];
    // function to create a download stream. streams are slightly delayed so that they will not end at the same time
    var testStream = function(i, delay) {
        setTimeout(
            function() {
                if (testState !== 1) return; // delayed stream ended up starting after the end of the download test
                tverb("dl test stream started " + i + " " + delay);
                var prevLoaded = 0; // number of bytes loaded last time onprogress was called
                var x = new XMLHttpRequest();
                xhr[i] = x;
                xhr[i].onprogress = function(event) {
                    tverb("dl stream progress event " + i + " " + event.loaded);
                    if (testState !== 1) {
                        try {
                            x.abort();
                        } catch (e) {}
                    } // just in case this XHR is still running after the download test
                    // progress event, add number of new loaded bytes to totLoaded
                    var loadDiff = event.loaded <= 0 ? 0 : event.loaded - prevLoaded;
                    if (isNaN(loadDiff) || !isFinite(loadDiff) || loadDiff < 0) return; // just in case
                    totLoaded += loadDiff;
                    prevLoaded = event.loaded;
                }.bind(this);
                xhr[i].onload = function() {
                    // the large file has been loaded entirely, start again
                    tverb("dl stream finished " + i);
                    try {
                        xhr[i].abort();
```

Figure 4.3 Algorithm of Download Speed (1)



```
                        xhr[i].abort();
                    } catch (e) {} // reset the stream data to empty ram
                    testStream(i, 0);
                }.bind(this);
                xhr[i].onerror = function() {
                    // error
                    tverb("dl stream failed " + i);
                    if (settings.xhr_ignoreErrors === 0) failed = true; //abort
                    try {
                        xhr[i].abort();
                    } catch (e) {}
                    delete xhr[i];
                    if (settings.xhr_ignoreErrors === 1) testStream(i, 0); //restart stream
                }.bind(this);
                // send xhr
                try {
                    if (settings.xhr_dlUseBlob) xhr[i].responseType = "blob";
                    else xhr[i].responseType = "arraybuffer";
                } catch (e) {}
                xhr[i].open("GET", settings.url_dl + url_sep(settings.url_dl) + (settings.mpot ? "cors=true&" : "") + "r=" + Math.random() + "&ckSize=" + settings.garbagePhp_chunkSize
true); // random string to prevent caching
                xhr[i].send();
            }.bind(this),
            1 + delay
        );
    }.bind(this);
    // open streams
    for (var i = 0; i < settings.xhr_dlMultistream; i++) {
        testStream(i, settings.xhr_multistreamDelay * i);
    }
    // every 200ms, update dlStatus
    interval = setInterval(
        function() {
            tverb("DL: " + dlStatus + (graceTimeDone ? "" : " (in grace time)"));
            var t = new Date().getTime() - startT;
            if (graceTimeDone) dlProgress = (t + bonusT) / (settings.time_dl_max * 1000);
            if (t < 200) return;
            if (!graceTimeDone) {
                if (t > 1000 * settings.time_dlGraceTime) {
                    if (totLoaded > 0) {
                        // if the connection is so slow that we didn't get a single chunk yet, do not reset
                        startT = new Date().getTime();
                        bonusT = 0;
```

Figure 4.4 Algorithm of Download Speed (2)



```
                        bonusT = 0;
                        totLoaded = 0.0;
                    }
                    graceTimeDone = true;
                }
            } else {
                var speed = totLoaded / (t / 1000.0);
                if (settings.time_auto) {
                    //decide how much to shorten the test. Every 200ms, the test is shortened by the bonusT calculated here
                    var bonus = (5.0 * speed) / 100000;
                    bonusT += bonus > 400 ? 400 : bonus;
                }
                //update status
                dlStatus = ((speed * 8 * settings.overheadCompensationFactor) / (settings.useMebibits ? 1048576 : 1000000)).toFixed(2); // speed is multiplied by 8 to go from bytes to bits, overhead compensation is applied, then everything is divided by 1048576 or 1000000 to go to megabits/mebibits
                if ((t + bonusT) / 1000.0 > settings.time_dl_max || failed) {
                    // test is over, stop streams and timer
                    if (failed || isNaN(dlStatus)) dlStatus = "Fail";
                    clearRequests();
                    clearInterval(interval);
                    dlProgress = 1;
                    tlog("dlTest: " + dlStatus + ", took " + (new Date().getTime() - startT) + "ms");
                    done();
                }
            }
        }.bind(this),
        200
    );
}
```

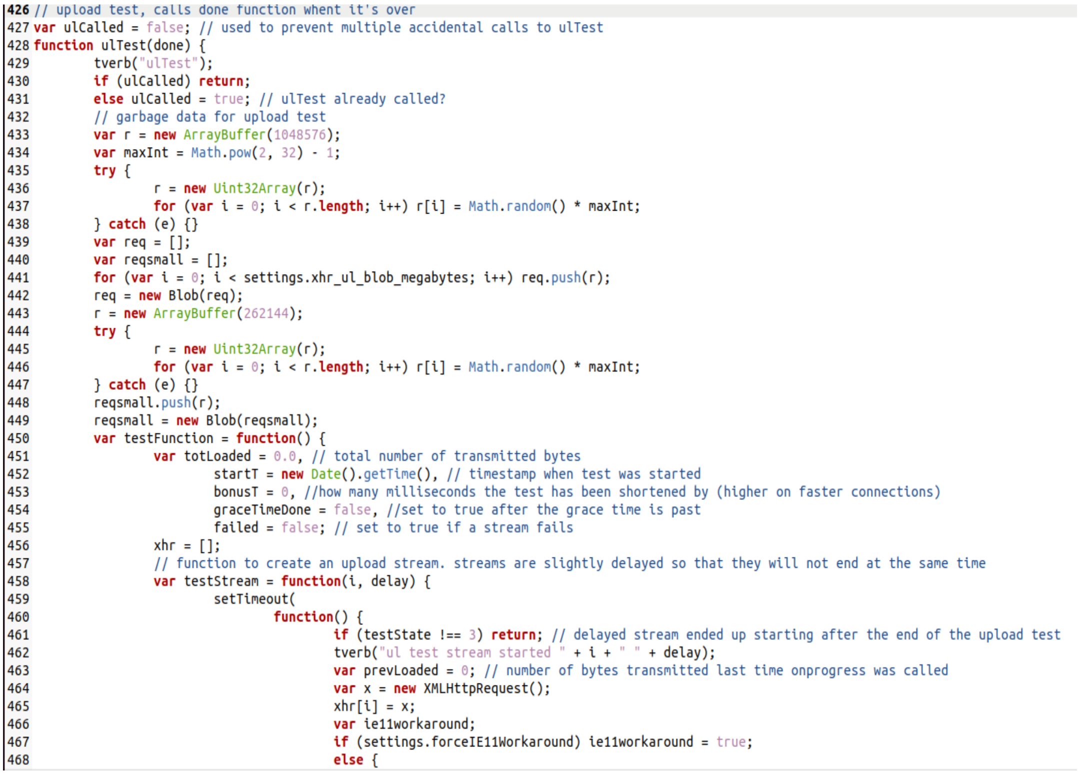
Figure 4.5 Algorithm of Download Speed (3)

This is a JavaScript function called "dlTest" that downloads a file from a specified URL and measures the download speed. The function takes one parameter, "done", which is a callback function that gets called when the download is complete. The function starts by initializing some variables, including "totLoaded", which keeps track of the total number of bytes loaded so far, "startT", which is the timestamp when the download started, and "failed", which is a flag that gets set to true if any of the download streams fail. The function then creates multiple "streams" to download the file. Each stream is slightly delayed so that they don't end at the same time. The streams are created using XMLHttpRequest, which is an API in JavaScript that allows you to send HTTP requests and receive responses.

The function also sets up event listeners for each stream, including an "onprogress" listener that gets called whenever new data is received. The "onprogress" listener adds the number of new bytes received to "totLoaded". The function then sets up a timer that updates the download status every 200 milliseconds. The download status includes the current download speed in megabits per second (Mbps). If the download speed is slower than a specified threshold, the function enters a "grace time" period where it waits for a certain amount of time to give the connection a chance to improve. If the connection is still slow after the grace time, the function ends the download and calls the "done" callback. If the download speed is fast enough, the function calculates a "bonus" time that is subtracted from the total download time to shorten the download test. This is done to prevent the download test from taking too long on fast connections. Finally, if the download test has run for longer than a specified maximum time, or if any of the download streams have failed, the function ends the download and calls the "done" callback.

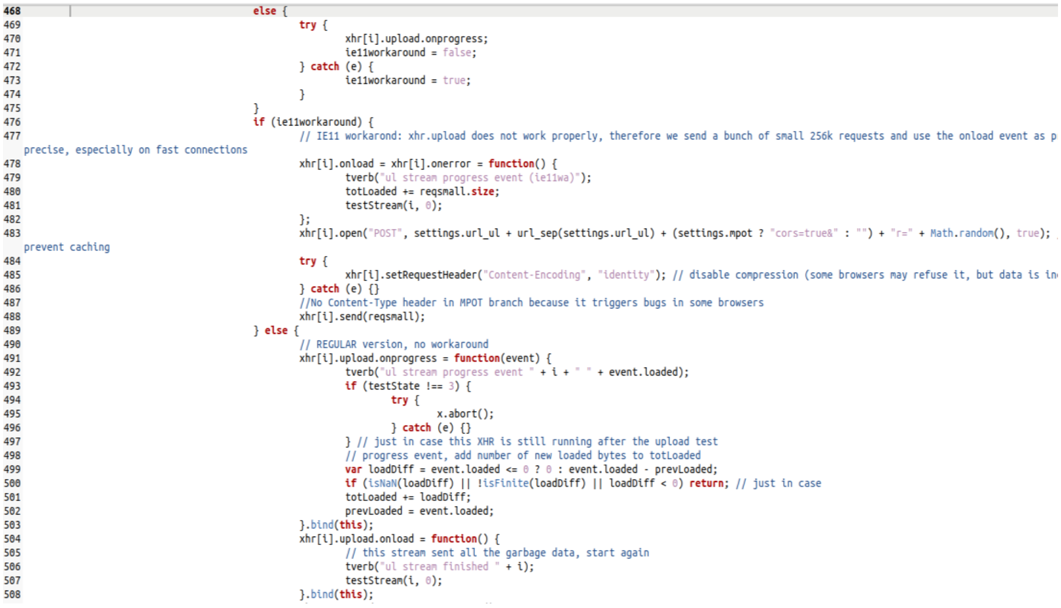
4.4.2 Upload Test Algorithm

The upload test algorithm measures the speed of data transfer from the client to the server. It works by uploading a file of a predetermined size to the server and measuring the time it takes to upload the file. The upload speed is then calculated by dividing the size of the file by the time taken to upload it.



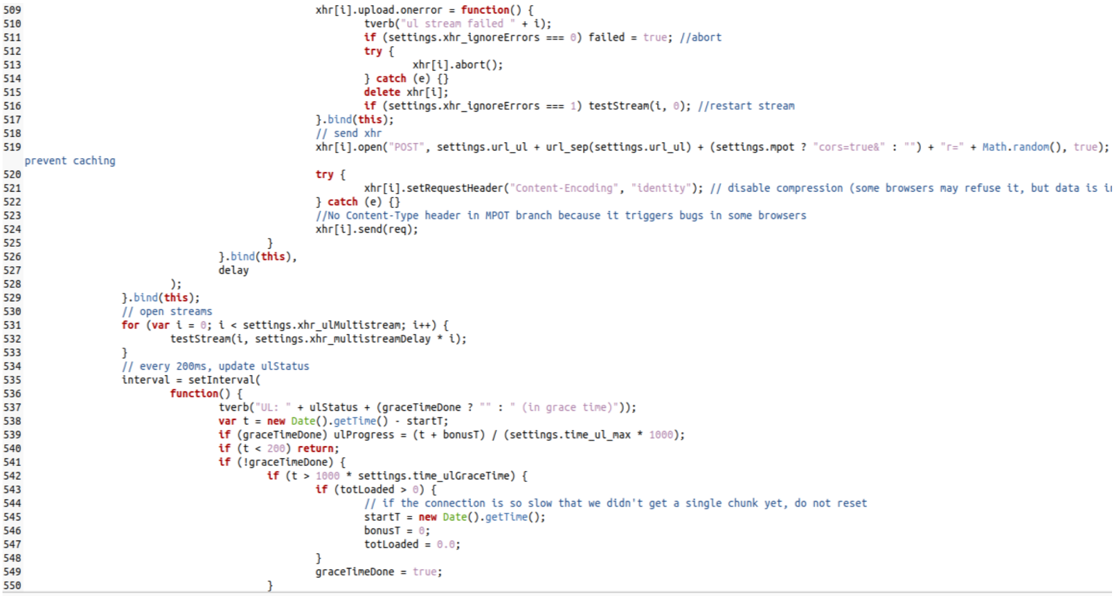
```
// upload test, calls done function whent it's over
var ulCalled = false; // used to prevent multiple accidental calls to ulTest
function ulTest(done) {
    tverb("ulTest");
    if (ulCalled) return;
    else ulCalled = true; // ulTest already called?
    // garbage data for upload test
    var r = new ArrayBuffer(1048576);
    var maxInt = Math.pow(2, 32) - 1;
    try {
        r = new Uint32Array(r);
        for (var i = 0; i < r.length; i++) r[i] = Math.random() * maxInt;
    } catch (e) {}
    var req = [];
    var reqsmall = [];
    for (var i = 0; i < settings.xhr_ul_blob_megabytes; i++) req.push(r);
    req = new Blob(req);
    r = new ArrayBuffer(262144);
    try {
        r = new Uint32Array(r);
        for (var i = 0; i < r.length; i++) r[i] = Math.random() * maxInt;
    } catch (e) {}
    reqsmall.push(r);
    reqsmall = new Blob(reqsmall);
    var testFunction = function() {
        var totLoaded = 0.0, // total number of transmitted bytes
            startT = new Date().getTime(), // timestamp when test was started
            bonusT = 0, //how many milliseconds the test has been shortened by (higher on faster connections)
            graceTimeDone = false, //set to true after the grace time is past
            failed = false; // set to true if a stream fails
        xhr = [];
        // function to create an upload stream. streams are slightly delayed so that they will not end at the same time
        var testStream = function(i, delay) {
            setTimeout(
                function() {
                    if (testState !== 3) return; // delayed stream ended up starting after the end of the upload test
                    tverb("ul test stream started " + i + " " + delay);
                    var prevLoaded = 0; // number of bytes transmitted last time onprogress was called
                    var x = new XMLHttpRequest();
                    xhr[i] = x;
                    var ie11workaround;
                    if (settings.forceIE11Workaround) ie11workaround = true;
                    else {
```

Figure 4.6 Algorithm of Upload Speed (1)



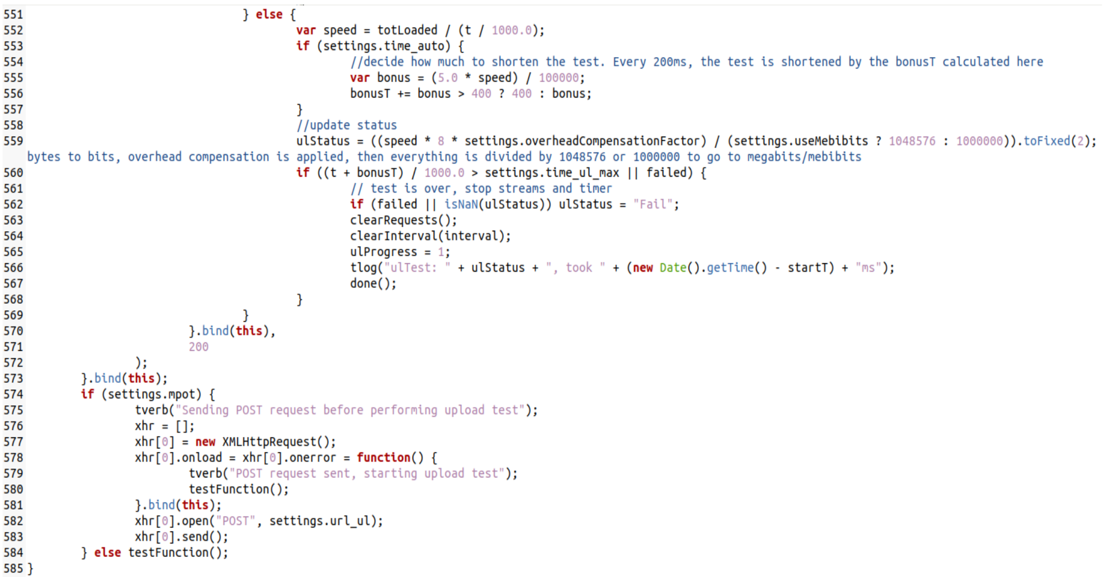
```
                    else {
                        try {
                            xhr[i].upload.onprogress;
                            ie11workaround = false;
                        } catch (e) {
                            ie11workaround = true;
                        }
                    }
                    if (ie11workaround) {
                        // IE11 workarond: xhr.upload does not work properly, therefore we send a bunch of small 256k requests and use the onload event as p
precise, especially on fast connections
                        xhr[i].onload = xhr[i].onerror = function() {
                            tverb("ul stream progress event (ie11wa)");
                            totLoaded += reqsmall.size;
                            testStream(i, 0);
                        };
                        xhr[i].open("POST", settings.url_ul + url_sep(settings.url_ul) + (settings.mpot ? "cors=true&" : "") + "r=" + Math.random(), true);
prevent caching
                        try {
                            xhr[i].setRequestHeader("Content-Encoding", "identity"); // disable compression (some browsers may refuse it, but data is in
                        } catch (e) {}
                        //No Content-Type header in MPOT branch because it triggers bugs in some browsers
                        xhr[i].send(reqsmall);
                    } else {
                        // REGULAR version, no workaround
                        xhr[i].upload.onprogress = function(event) {
                            tverb("ul stream progress event " + i + " " + event.loaded);
                            if (testState !== 3) {
                                try {
                                    x.abort();
                                } catch (e) {}
                            } // just in case this XHR is still running after the upload test
                            // progress event, add number of new loaded bytes to totLoaded
                            var loadDiff = event.loaded <= 0 ? 0 : event.loaded - prevLoaded;
                            if (isNaN(loadDiff) || !isFinite(loadDiff) || loadDiff < 0) return; // just in case
                            totLoaded += loadDiff;
                            prevLoaded = event.loaded;
                        }.bind(this);
                        xhr[i].upload.onload = function() {
                            // this stream sent all the garbage data, start again
                            tverb("ul stream finished " + i);
                            testStream(i, 0);
                        }.bind(this);
```

Figure 4.7 Algorithm of Upload Speed (2)



```
                        xhr[i].upload.onerror = function() {
                            tverb("ul stream failed " + i);
                            if (settings.xhr_ignoreErrors === 0) failed = true; //abort
                            try {
                                xhr[i].abort();
                            } catch (e) {}
                            delete xhr[i];
                            if (settings.xhr_ignoreErrors === 1) testStream(i, 0); //restart stream
                        }.bind(this);
                        // send xhr
                        xhr[i].open("POST", settings.url_ul + url_sep(settings.url_ul) + (settings.mpot ? "cors=true&" : "") + "r=" + Math.random(), true); //
prevent caching
                        try {
                            xhr[i].setRequestHeader("Content-Encoding", "identity"); // disable compression (some browsers may refuse it, but data is i
                        } catch (e) {}
                        //No Content-Type header in MPOT branch because it triggers bugs in some browsers
                        xhr[i].send(req);
                    }
                }.bind(this),
                delay
            );
        }.bind(this);
        // open streams
        for (var i = 0; i < settings.xhr_ulMultistream; i++) {
            testStream(i, settings.xhr_multistreamDelay * i);
        }
        // every 200ms, update ulStatus
        interval = setInterval(
            function() {
                tverb("UL: " + ulStatus + (graceTimeDone ? "" : " (in grace time)"));
                var t = new Date().getTime() - startT;
                if (graceTimeDone) ulProgress = (t + bonusT) / (settings.time_ul_max * 1000);
                if (t < 200) return;
                if (!graceTimeDone) {
                    if (t > 1000 * settings.time_ulGraceTime) {
                        if (totLoaded > 0) {
                            // if the connection is so slow that we didn't get a single chunk yet, do not reset
                            startT = new Date().getTime();
                            bonusT = 0;
                            totLoaded = 0.0;
                        }
                        graceTimeDone = true;
                    }
```

Figure 4.8 Algorithm of Upload Speed (3)



```
                    } else {
                        var speed = totLoaded / (t / 1000.0);
                        if (settings.time_auto) {
                            //decide how much to shorten the test. Every 200ms, the test is shortened by the bonusT calculated here
                            var bonus = (5.0 * speed) / 100000;
                            bonusT += bonus > 400 ? 400 : bonus;
                        }
                        //update status
                        ulStatus = ((speed * 8 * settings.overheadCompensationFactor) / (settings.useMebibits ? 1048576 : 1000000)).toFixed(2); //
bytes to bits, overhead compensation is applied, then everything is divided by 1048576 or 1000000 to go to megabits/mebibits
                        if ((t + bonusT) / 1000.0 > settings.time_ul_max || failed) {
                            // test is over, stop streams and timer
                            if (failed || isNaN(ulStatus)) ulStatus = "Fail";
                            clearRequests();
                            clearInterval(interval);
                            ulProgress = 1;
                            tlog("ulTest: " + ulStatus + ", took " + (new Date().getTime() - startT) + "ms");
                            done();
                        }
                    }
                }.bind(this),
                200
            );
        }.bind(this);
        if (settings.mpot) {
            tverb("Sending POST request before performing upload test");
            xhr = [];
            xhr[0] = new XMLHttpRequest();
            xhr[0].onload = xhr[0].onerror = function() {
                tverb("POST request sent, starting upload test");
                testFunction();
            }.bind(this);
            xhr[0].open("POST", settings.url_ul);
            xhr[0].send();
        } else testFunction();
    }
```

Figure 4.9 Algorithm of Upload Speed (4)

This script appears to be a test function for uploading data to a server. The function takes a done function as a parameter which is called when the test has completed.

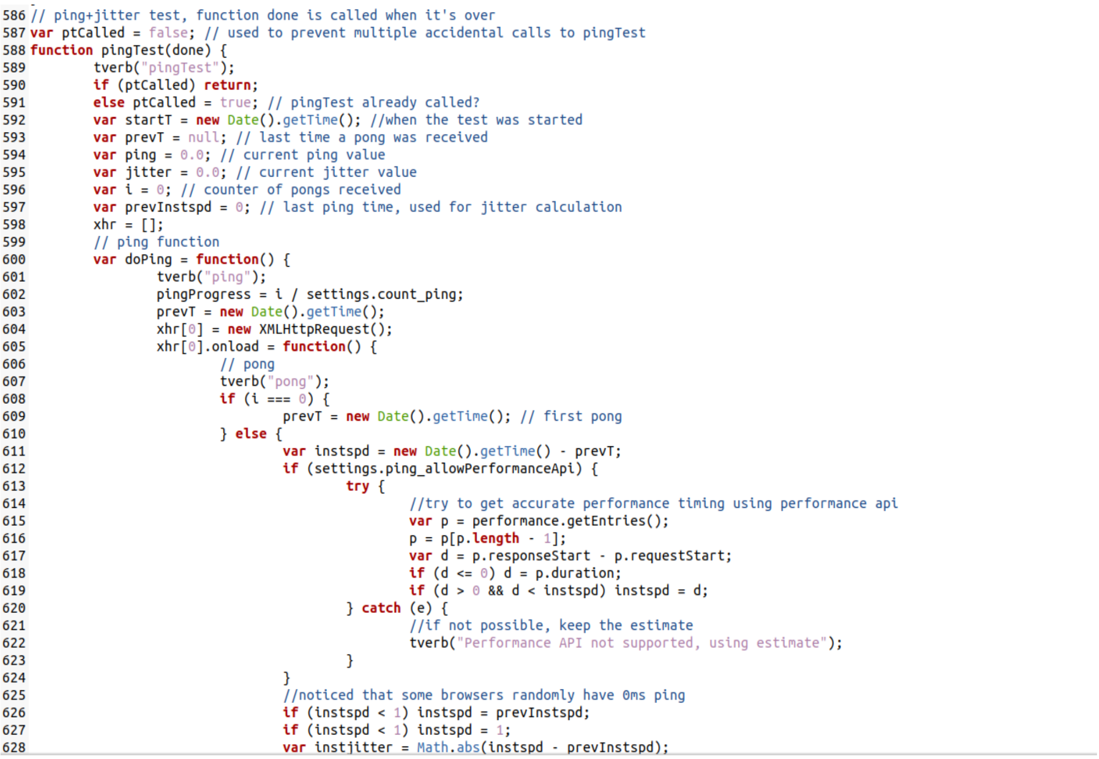
In the function, a large amount of garbage data is generated, and then uploaded to the server using multiple XHR (XMLHttpRequest) requests. There are two types of XHR requests used in the function: ie11workaround and regular. The former is used when the settings.forceIE11Workaround flag is set to true, and the latter is used otherwise. The regular XHR requests use the onprogress, onload, and onerror events to track the progress of the upload and handle errors. The ie11workaround XHR requests use only the onload and onerror events to track the progress of the upload because the xhr.upload.onprogress event is not available in IE11.

The progress of each XHR request is tracked using the totLoaded variable, which is incremented each time the onprogress event fires. The graceTimeDone flag is used to determine when the test has completed. The function first creates multiple req and reqsmall Blobs containing the garbage data to be uploaded. It then creates settings.xhr_ul_blob_megabytes req Blobs and one reqsmall Blob. These are used in the XHR requests to upload the data to the server. The testStream function is called for each of the req Blobs. Each testStream function creates a new XHR object, sets up the appropriate event listeners based on the type of XHR request being used, and sends the XHR object with the send method. When the onload or onerror event fires, the testStream function is called again to upload more data. If the ie11workaround flag is set, reqsmall Blobs are used instead of req Blobs.

The testFunction function sets the initial values of the variables used to track the progress of the upload and starts the testStream function for each req Blob. It also starts a timer to keep track of how long the test has been running. When the timer has elapsed for settings.graceTimeMs milliseconds, the graceTimeDone flag is set to true. When all of the XHR requests have completed, the done function is called. If any of the XHR requests fail, the failed flag is set to true. If the xhr_ignoreErrors flag is set to 0, the test is aborted. If it is set to 1, the testStream function is called again to retry the upload.

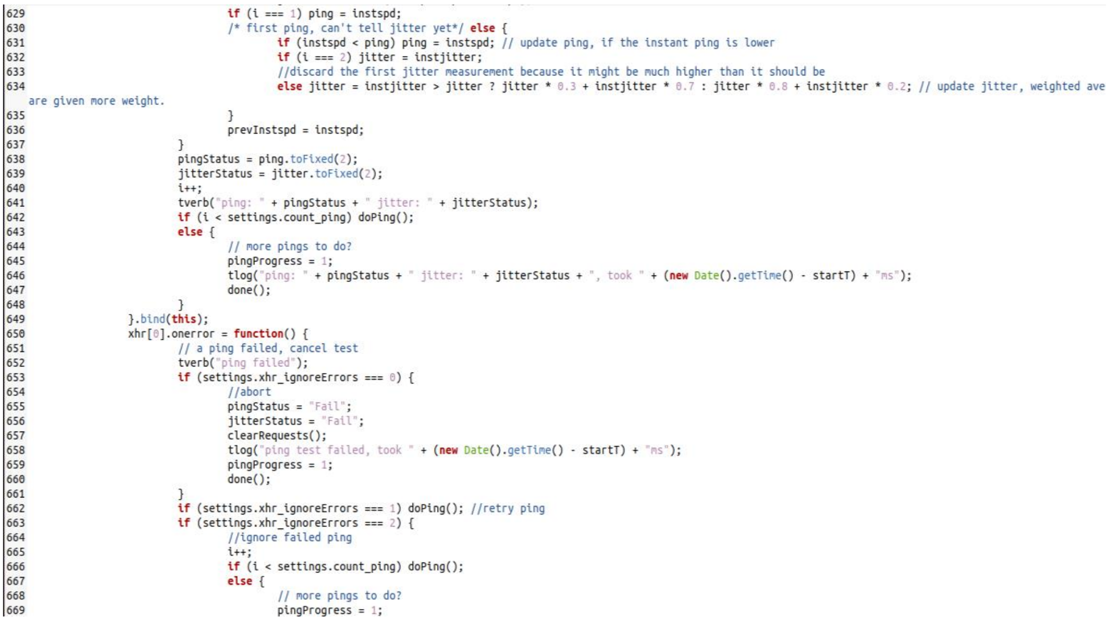
4.4.3 Ping and Jitter Test Algorithm

The ping test algorithm measures the time it takes for a small packet of data to travel from the client to the server and back. It works by sending a small packet of data to the server and measuring the time it takes to receive a response. The ping time is a measure of the latency or delay in the network. On the other hand, the jitter test algorithm measures the variability in the latency of the network. It works by sending multiple small packets of data to the server and measuring the difference in the time it takes for each packet to travel from the client to the server and back. A high jitter value indicates a network with inconsistent latency.



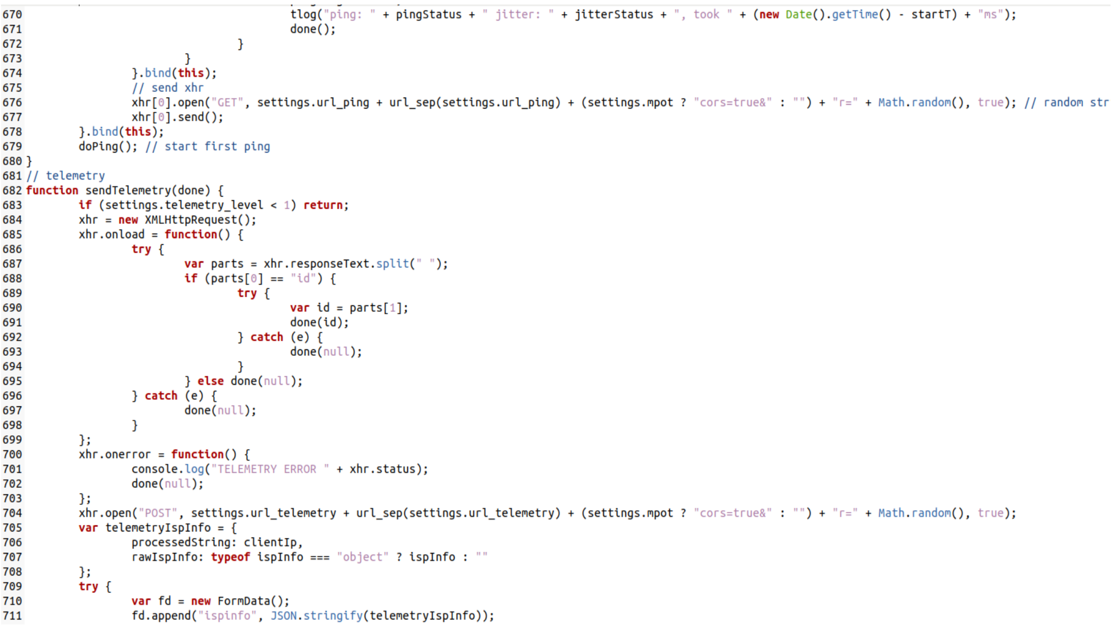
```
// ping+jitter test, function done is called when it's over
var ptCalled = false; // used to prevent multiple accidental calls to pingTest
function pingTest(done) {
    tverb("pingTest");
    if (ptCalled) return;
    else ptCalled = true; // pingTest already called?
    var startT = new Date().getTime(); //when the test was started
    var prevT = null; // last time a pong was received
    var ping = 0.0; // current ping value
    var jitter = 0.0; // current jitter value
    var i = 0; // counter of pongs received
    var prevInstspd = 0; // last ping time, used for jitter calculation
    xhr = [];
    // ping function
    var doPing = function() {
        tverb("ping");
        pingProgress = i / settings.count_ping;
        prevT = new Date().getTime();
        xhr[0] = new XMLHttpRequest();
        xhr[0].onload = function() {
            // pong
            tverb("pong");
            if (i === 0) {
                prevT = new Date().getTime(); // first pong
            } else {
                var instspd = new Date().getTime() - prevT;
                if (settings.ping_allowPerformanceApi) {
                    try {
                        //try to get accurate performance timing using performance api
                        var p = performance.getEntries();
                        p = p[p.length - 1];
                        var d = p.responseStart - p.requestStart;
                        if (d <= 0) d = p.duration;
                        if (d > 0 && d < instspd) instspd = d;
                    } catch (e) {
                        //if not possible, keep the estimate
                        tverb("Performance API not supported, using estimate");
                    }
                }
                //noticed that some browsers randomly have 0ms ping
                if (instspd < 1) instspd = prevInstspd;
                if (instspd < 1) instspd = 1;
                var instjitter = Math.abs(instspd - prevInstspd);
```

Figure 4.10 Algorithm of Ping&Jitter Speed (1)



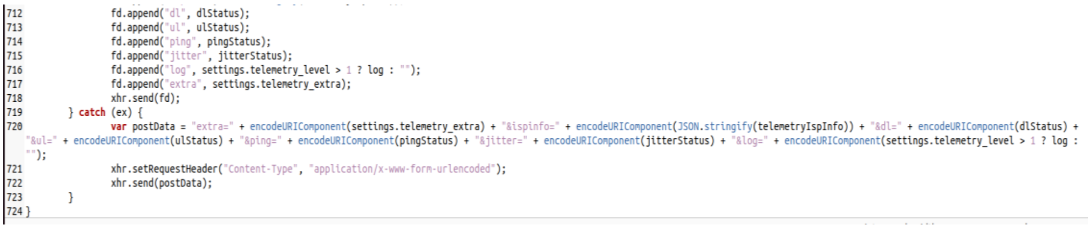
```
                if (i === 1) ping = instspd;
                /* first ping, can't tell jitter yet*/ else {
                    if (instspd < ping) ping = instspd; // update ping, if the instant ping is lower
                    if (i === 2) jitter = instjitter;
                    //discard the first jitter measurement because it might be much higher than it should be
                    else jitter = instjitter > jitter ? jitter * 0.3 + instjitter * 0.7 : jitter * 0.8 + instjitter * 0.2; // update jitter, weighted ave
are given more weight.
                }
                prevInstspd = instspd;
            }
            pingStatus = ping.toFixed(2);
            jitterStatus = jitter.toFixed(2);
            i++;
            tverb("ping: " + pingStatus + " jitter: " + jitterStatus);
            if (i < settings.count_ping) doPing();
            else {
                // more pings to do?
                pingProgress = 1;
                tlog("ping: " + pingStatus + " jitter: " + jitterStatus + ", took " + (new Date().getTime() - startT) + "ms");
                done();
            }
        }.bind(this);
        xhr[0].onerror = function() {
            // a ping failed, cancel test
            tverb("ping failed");
            if (settings.xhr_ignoreErrors === 0) {
                //abort
                pingStatus = "Fail";
                jitterStatus = "Fail";
                clearRequests();
                tlog("ping test failed, took " + (new Date().getTime() - startT) + "ms");
                pingProgress = 1;
                done();
            }
            if (settings.xhr_ignoreErrors === 1) doPing(); //retry ping
            if (settings.xhr_ignoreErrors === 2) {
                //ignore failed ping
                i++;
                if (i < settings.count_ping) doPing();
                else {
                    // more pings to do?
                    pingProgress = 1;
```

Figure 4.11 Algorithm of Ping&Jitter Speed (2)



```
                    tlog("ping: " + pingStatus + " jitter: " + jitterStatus + ", took " + (new Date().getTime() - startT) + "ms");
                    done();
                }
            }
        }.bind(this);
        // send xhr
        xhr[0].open("GET", settings.url_ping + url_sep(settings.url_ping) + (settings.mpot ? "cors=true&" : "") + "r=" + Math.random(), true); // random str
        xhr[0].send();
    }.bind(this);
    doPing(); // start first ping
}
// telemetry
function sendTelemetry(done) {
    if (settings.telemetry_level < 1) return;
    xhr = new XMLHttpRequest();
    xhr.onload = function() {
        try {
            var parts = xhr.responseText.split(" ");
            if (parts[0] == "id") {
                try {
                    var id = parts[1];
                    done(id);
                } catch (e) {
                    done(null);
                }
            } else done(null);
        } catch (e) {
            done(null);
        }
    };
    xhr.onerror = function() {
        console.log("TELEMETRY ERROR " + xhr.status);
        done(null);
    };
    xhr.open("POST", settings.url_telemetry + url_sep(settings.url_telemetry) + (settings.mpot ? "cors=true&" : "") + "r=" + Math.random(), true);
    var telemetryIspInfo = {
        processedString: clientIp,
        rawIspInfo: typeof ispInfo === "object" ? ispInfo : ""
    };
    try {
        var fd = new FormData();
        fd.append("ispinfo", JSON.stringify(telemetryIspInfo));
```

Figure 4.12 Algorithm of Ping&Jitter Speed (3)



```
        fd.append("dl", dlStatus);
        fd.append("ul", ulStatus);
        fd.append("ping", pingStatus);
        fd.append("jitter", jitterStatus);
        fd.append("log", settings.telemetry_level > 1 ? log : "");
        fd.append("extra", settings.telemetry_extra);
        xhr.send(fd);
    } catch (ex) {
        var postData = "extra=" + encodeURIComponent(settings.telemetry_extra) + "&ispinfo=" + encodeURIComponent(JSON.stringify(telemetryIspInfo)) + "&dl=" + encodeURIComponent(dlStatus) +
"&ul=" + encodeURIComponent(ulStatus) + "&ping=" + encodeURIComponent(pingStatus) + "&jitter=" + encodeURIComponent(jitterStatus) + "&log=" + encodeURIComponent(settings.telemetry_level > 1 ? log :
"");
        xhr.setRequestHeader("Content-Type", "application/x-www-form-urlencoded");
        xhr.send(postData);
    }
}
```

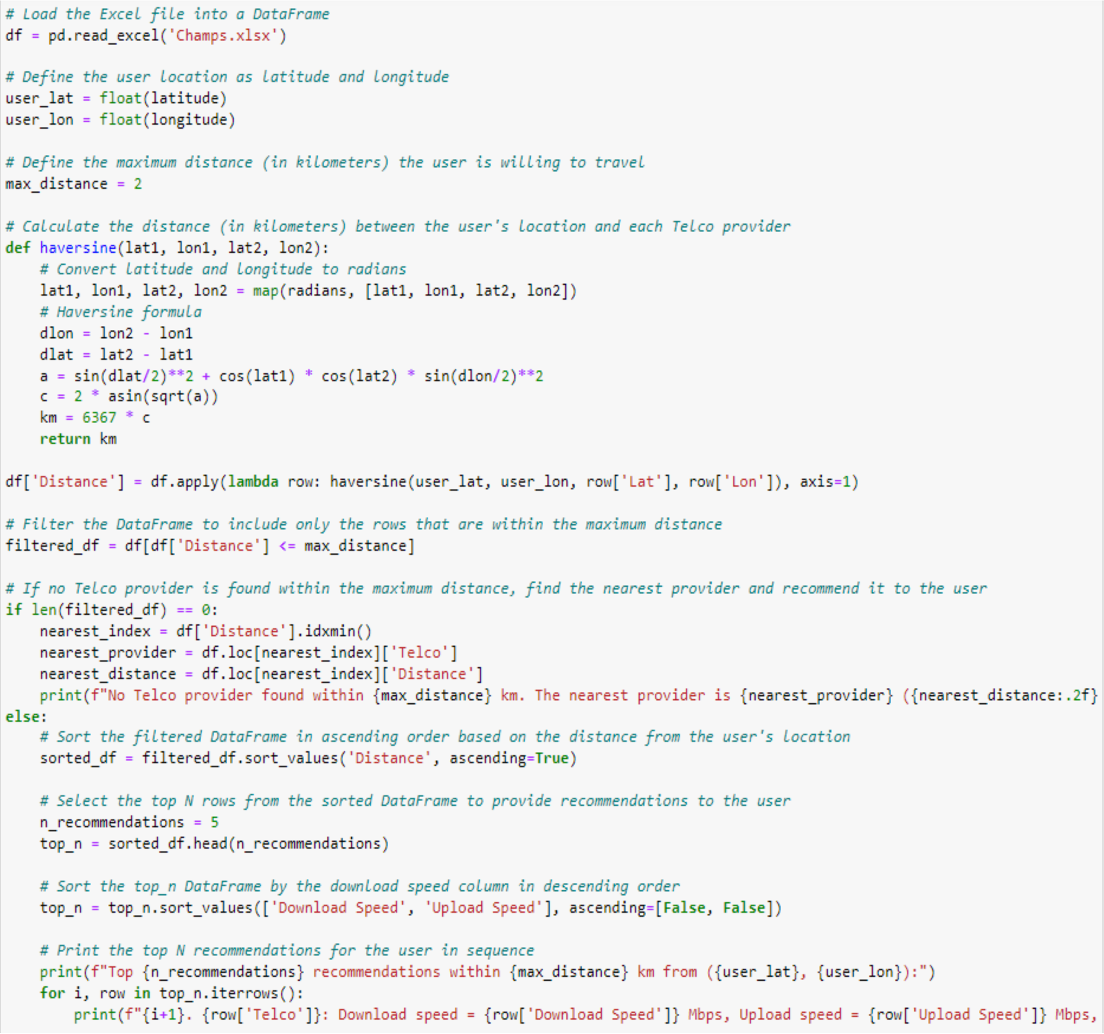
Figure 4.13 Algorithm of Ping&Jitter Speed (4)

The code appears to be JavaScript code for performing a ping and jitter test and sending telemetry. The ping and jitter test function is named pingTest and takes a callback function done as a parameter, which is called when the test is over. The sendTelemetry function sends telemetry data and also takes a callback function done as a parameter, which is called when the telemetry data has been sent.

The pingTest function uses an XMLHttpRequest object to send a GET request to a specified URL with a random string appended to prevent caching. The onload and onerror event listeners are used to handle the response from the server. The prevT variable is used to store the time of the last received "pong" response from the server. The ping and jitter variables store the current ping and jitter values. The instspd variable stores the time between the previous and current "pong" responses. The prevInstspd variable stores the previous instspd value for use in calculating the jitter. The i variable is a counter for the number of "pong" responses received. The pingProgress variable is used to track the progress of the ping test. The settings object contains various settings for the test.

The sendTelemetry function also uses an XMLHttpRequest object to send a POST request with telemetry data to a specified URL. The onload and onerror event listeners are used to handle the response from the server. The settings object contains various settings for the telemetry.

4.4.4 Telco Recommendation Algorithm



```
# Load the Excel file into a DataFrame
df = pd.read_excel('Champs.xlsx')

# Define the user location as latitude and longitude
user_lat = float(latitude)
user_lon = float(longitude)

# Define the maximum distance (in kilometers) the user is willing to travel
max_distance = 2

# Calculate the distance (in kilometers) between the user's location and each Telco provider
def haversine(lat1, lon1, lat2, lon2):
    # Convert latitude and longitude to radians
    lat1, lon1, lat2, lon2 = map(radians, [lat1, lon1, lat2, lon2])
    # Haversine formula
    dlon = lon2 - lon1
    dlat = lat2 - lat1
    a = sin(dlat/2)**2 + cos(lat1) * cos(lat2) * sin(dlon/2)**2
    c = 2 * asin(sqrt(a))
    km = 6367 * c
    return km

df['Distance'] = df.apply(lambda row: haversine(user_lat, user_lon, row['Lat'], row['Lon']), axis=1)

# Filter the DataFrame to include only the rows that are within the maximum distance
filtered_df = df[df['Distance'] <= max_distance]

# If no Telco provider is found within the maximum distance, find the nearest provider and recommend it to the user
if len(filtered_df) == 0:
    nearest_index = df['Distance'].idxmin()
    nearest_provider = df.loc[nearest_index]['Telco']
    nearest_distance = df.loc[nearest_index]['Distance']
    print(f"No Telco provider found within {max_distance} km. The nearest provider is {nearest_provider} ({nearest_distance:.2f}
else:
    # Sort the filtered DataFrame in ascending order based on the distance from the user's location
    sorted_df = filtered_df.sort_values('Distance', ascending=True)

    # Select the top N rows from the sorted DataFrame to provide recommendations to the user
    n_recommendations = 5
    top_n = sorted_df.head(n_recommendations)

    # Sort the top_n DataFrame by the download speed column in descending order
    top_n = top_n.sort_values(['Download Speed', 'Upload Speed'], ascending=[False, False])

    # Print the top N recommendations for the user in sequence
    print(f"Top {n_recommendations} recommendations within {max_distance} km from ({user_lat}, {user_lon}):")
    for i, row in top_n.iterrows():
        print(f"{i+1}. {row['Telco']}: Download speed = {row['Download Speed']} Mbps, Upload speed = {row['Upload Speed']} Mbps,
```

Figure 4.14 Algorithm of Telco Recommendation

The algorithm in the code above is designed to recommend the top Telco providers for a user based on their location, download speed, upload speed, and latency. The algorithm utilizes the Haversine formula to calculate the distance between the user's location (specified by the user's latitude and longitude) and the location of each Telco provider in the dataset. The maximum distance that the user is willing to travel is also specified as a parameter in the code.

The code first loads an Excel file containing information about the Telco providers into a panda DataFrame. The user's latitude and longitude are then defined, and the maximum distance that the user is willing to travel is specified. The Haversine function is used to calculate the distance (in kilometers) between the user's location and each Telco provider's location, and this information is stored in a new column called 'Distance' in the DataFrame.

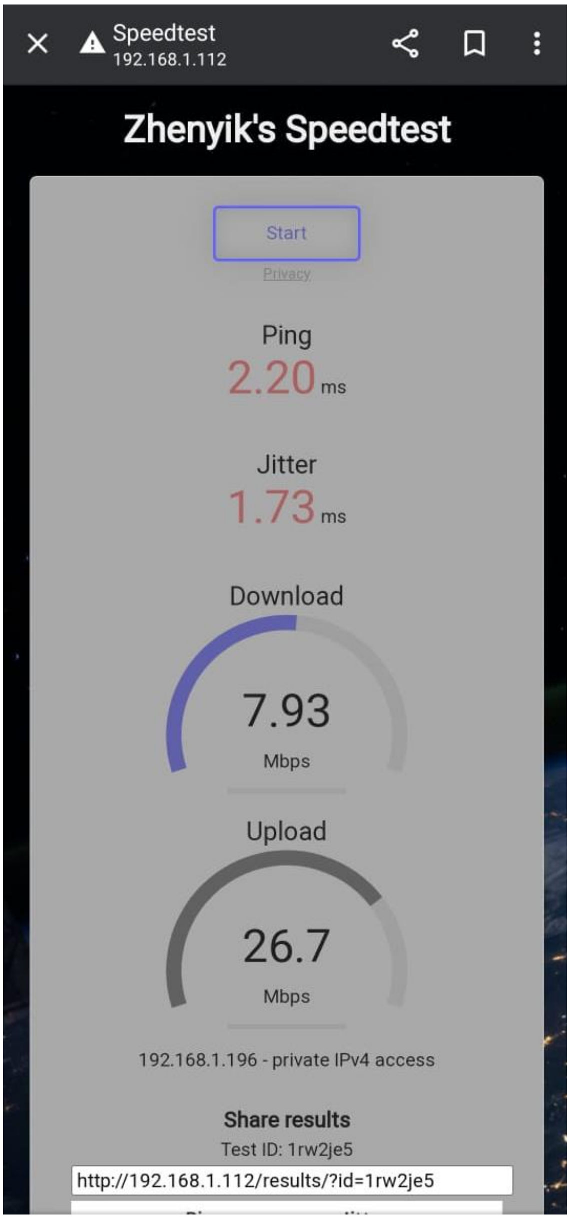
The DataFrame is then filtered to include only the Telco providers that are within the maximum distance specified by the user. If there are no Telco providers found within the maximum distance, the algorithm recommends the nearest provider to the user based on their location. Otherwise, the DataFrame is sorted in ascending order based on the distance from the user's location.

The top N rows from the filtered and sorted DataFrame are then selected to provide recommendations to the user, where N is specified as a parameter in the code. The top N rows are sorted by the download speed column in descending order, and then by the upload speed column in descending order as a tiebreaker. Finally, the top N recommendations are printed in sequence, with each row displaying the Telco provider's name, download speed, upload speed, and latency.

Overall, this algorithm is a useful tool for users who are looking for Telco providers within a certain distance from their location and who prioritize download and upload speeds. By using the Haversine formula to calculate distances and sorting Telco providers by download and upload speeds, the algorithm can provide accurate and relevant recommendations to users.

4.5 Graphical User Interface Design

4.5.1 Speedtest HomePage



Zhenyik's Speedtest

Ping
2.20 ms

Jitter
1.73 ms

Download
7.93
Mbps

Upload
26.7
Mbps

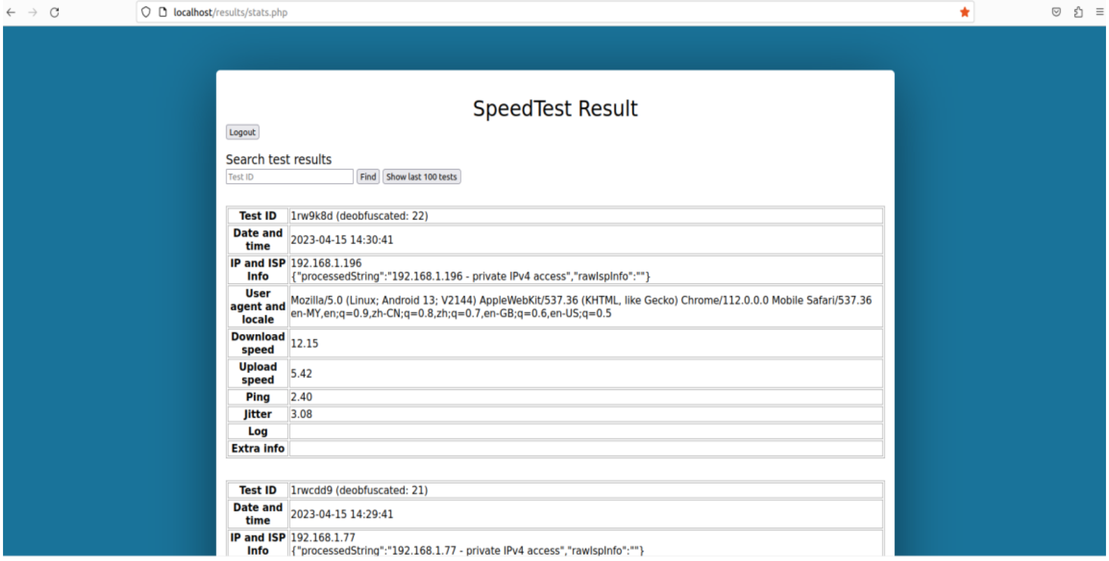
192.168.1.196 - private IPv4 access

Share results
Test ID: 1rw2je5
http://192.168.1.112/results/?id=1rw2je5

Figure 4.15 ZhenYik’s SpeedTest Home Page

The interface for a speed test application should be simple and intuitive, allowing users to quickly and easily initiate a test to measure their internet connection speed. The main elements of such an interface would include a large, clearly visible "Start" button that initiates the test, along with a progress bar that visually displays the progress of the test. The interface should also provide numerical displays of the user's download and upload speeds, so that users can see their results quickly and easily. Users should be able to select the server location from a list of available servers to ensure that they are testing their connection speed to the appropriate location.

4.5.2 Speedtest Result Page



localhost/results/stats.php

SpeedTest Result

Search test results

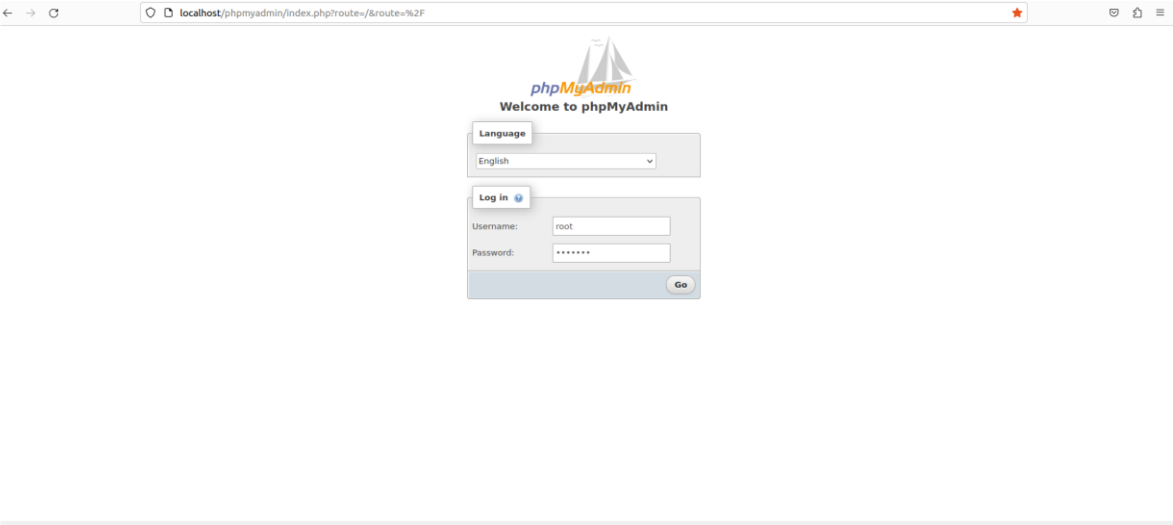
Test ID	1rw9k8d (deobfuscated: 22)
Date and time	2023-04-15 14:30:41
IP and ISP Info	192.168.1.196 {"processedString":"192.168.1.196 - private IPv4 access","rawIspInfo":""}
User agent and locale	Mozilla/5.0 (Linux; Android 13; V2144) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.0.0 Mobile Safari/537.36 en-MY,en;q=0.9,zh-CN;q=0.8,zh;q=0.7,en-GB;q=0.6,en-US;q=0.5
Download speed	12.15
Upload speed	5.42
Ping	2.40
Jitter	3.08
Log	
Extra info	

Test ID	1rwcdd9 (deobfuscated: 21)
Date and time	2023-04-15 14:29:41
IP and ISP Info	192.168.1.77 {"processedString":"192.168.1.77 - private IPv4 access","rawIspInfo":""}

Figure 4.16 ZhenYik's SpeedTest Result History Page

This is the user interface of the speed test result, and it will include the Test ID, Data and time, IP and ISP, User agent and locale, Download speed, Upload Speed, Ping, and jitter. Each speed test result will have their own unique Test ID, and user can search the Test ID through the search bar. And all of the data is stored inside the MySQL Ubuntu and use the PhpMyAdmin to manage.

4.5.3 PhpMyadmin Page



localhost/phpmyadmin/index.php?route=/&route=%2F

Welcome to phpMyAdmin

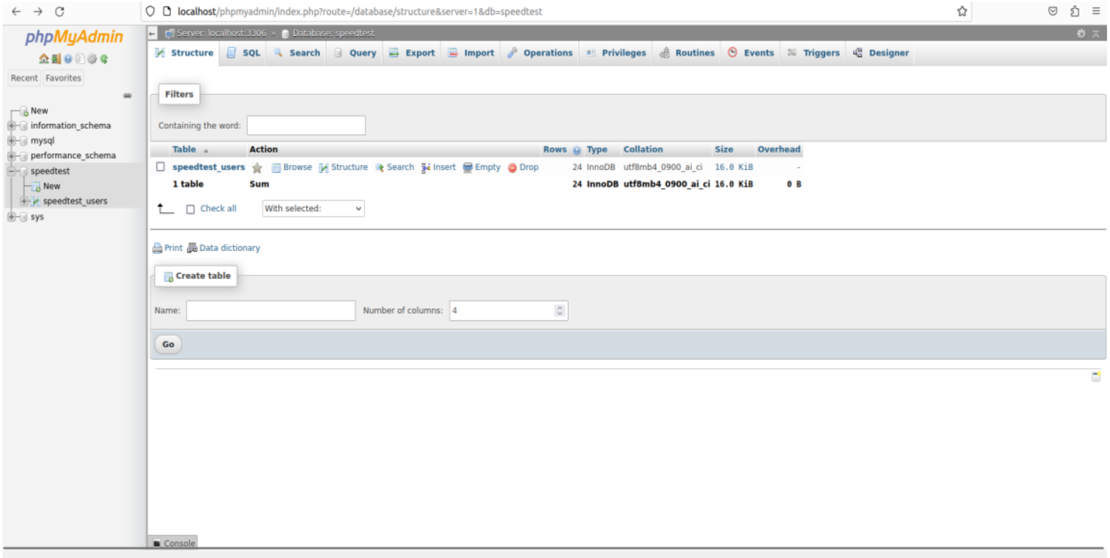
Language: English

Log in

Username: root

Password: •••••••

Figure 4.17 PhpMyAdmin Login Page



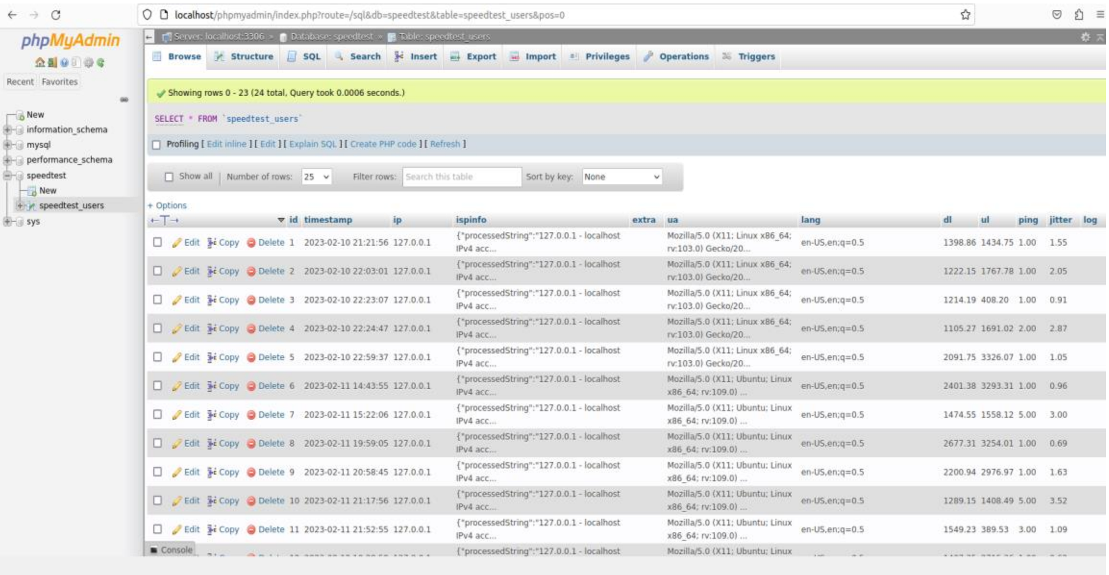
Containing the word:

Table	Action	Rows	Type	Collation	Size	Overhead
□ speedtest_users	Browse Structure Search Insert Empty Drop	24	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	-
1 table	Sum	24	InnoDB	utf8mb4_0900_ai_ci	16.0 KiB	0 B

□ Check all With selected:

Name: Number of columns: 4

Figure 4.18 PhpMyAdmin Home Page



Showing rows 0 - 23 (24 total, Query took 0.0006 seconds.)

```
SELECT * FROM `speedtest_users`
```

□ Profiling [Edit inline] [Edit] [Explain SQL] [Create PHP code] [Refresh]

□ Show all | Number of rows: 25 Filter rows: Search this table Sort by key: None

	id	timestamp	ip	ispinfo	extra	ua	lang	dl	ul	ping	jitter	log
□ Edit Copy Delete	1	2023-02-10 21:21:56	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Linux x86_64; rv:103.0) Gecko/20...	en-US,en;q=0.5	1398.86	1434.75	1.00	1.55	
□ Edit Copy Delete	2	2023-02-10 22:03:01	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Linux x86_64; rv:103.0) Gecko/20...	en-US,en;q=0.5	1222.15	1767.78	1.00	2.05	
□ Edit Copy Delete	3	2023-02-10 22:23:07	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Linux x86_64; rv:103.0) Gecko/20...	en-US,en;q=0.5	1214.19	408.20	1.00	0.91	
□ Edit Copy Delete	4	2023-02-10 22:24:47	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Linux x86_64; rv:103.0) Gecko/20...	en-US,en;q=0.5	1105.27	1691.02	2.00	2.87	
□ Edit Copy Delete	5	2023-02-10 22:59:37	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Linux x86_64; rv:103.0) Gecko/20...	en-US,en;q=0.5	2091.75	3326.07	1.00	1.05	
□ Edit Copy Delete	6	2023-02-11 14:43:55	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:109.0) ...	en-US,en;q=0.5	2401.38	3293.31	1.00	0.96	
□ Edit Copy Delete	7	2023-02-11 15:22:06	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:109.0) ...	en-US,en;q=0.5	1474.55	1558.12	5.00	3.00	
□ Edit Copy Delete	8	2023-02-11 19:59:05	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:109.0) ...	en-US,en;q=0.5	2677.31	3254.01	1.00	0.69	
□ Edit Copy Delete	9	2023-02-11 20:58:45	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:109.0) ...	en-US,en;q=0.5	2200.94	2976.97	1.00	1.63	
□ Edit Copy Delete	10	2023-02-11 21:17:56	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:109.0) ...	en-US,en;q=0.5	1289.15	1408.49	5.00	3.52	
□ Edit Copy Delete	11	2023-02-11 21:52:55	127.0.0.1	{"processedString":"127.0.0.1 - localhost IPv4 acc...		Mozilla/5.0 (X11; Ubuntu; Linux x86_64; rv:109.0) ...	en-US,en;q=0.5	1549.23	389.53	3.00	1.09	

Figure 4.19 PhpMyAdmin Result Management Page

The menu bar of PhpMyAdmin is located at the top of the interface and provides access to a range of useful features. These include the ability to create new databases, tables, and fields, as well as to import and export data from the database. The menu bar also provides access to the SQL editor, which allows users to write and execute custom SQL queries. Additionally, there are options to manage user accounts, set up server connections, and configure various settings for the interface. The menu bar is organized into several sections, such as the home section, which provides an overview of the current database system, and the Operations section, which offers tools for optimizing, repairing, and analyzing databases. Overall, the menu bar is an essential component of the PhpMyAdmin user interface, offering easy access to a range of powerful features for managing MySQL and MariaDB databases.

4.6 Concluding Remark

In conclusion, this system design involved using Jupyter Notebook for data analysis and processing, integrating the data with a web server, and utilizing the internet to access external resources. Additionally, the algorithm design was critical to the success of the project, and we used several algorithms such as the haversine algorithm to calculate distances, as well as algorithms to measure download and upload speeds, ping, and jitter.

The graphical user interface design was also essential to the project, as it allowed end-users to interact with the system and visualize data in an intuitive and easy-to-use way. By creating a clean and simple interface, we were able to ensure that users could access the insights they needed quickly and efficiently.

Overall, this project has demonstrated the power of data analytics and visualization in providing valuable insights and recommendations to users. By collecting and aggregating data, analyzing it using sophisticated algorithms, and presenting it in a user-friendly interface, we have enabled users to make informed decisions about their telecommunications needs. Going forward, it will be essential to continue testing and maintaining the system to ensure that it remains up-to-date and effective.

In summary, this project has been an exciting and challenging endeavour that has demonstrated the power of data analytics and visualization in providing valuable insights and recommendations to end-users. By incorporating a range of components, from data collection to user interface design, we have created a robust and effective system that can help users make informed decisions about their telecommunications needs. With ongoing testing and maintenance, this system will continue to provide value to users for years to come.

CHAPTER 5

5.1 Setting UP

5.1.1 Hardware

5.1.1.1 Below RM50 (Prepaid)

In order to prevent unfair comparison occurs when I was collecting the data, I did the research for the package provided by each telco, below are my research result:

Table 5.1 Prepaid Plan for Each Telco

	Name of Plans	*Speed Cap*	*Prices*	*Capacity*
DIGI	Raja Kombo 45	Uncapped	RM45	UNLIMITED
Maxis	RM45/mth	6Mbps	RM45	UNLIMITED
U-mobile	Brand H	6Mbps	RM45	UNLIMITED
Celcom	Unlimited RM45	6Mbps	RM45	UNLIMITED
Yes	Yes Kasi Up Prepaid 30	7Mbps	RM30	UNLIMITED

Above are the most expensive prepaid plan provided by each telco, but most of the plans have a network speed cap, so this project will not use prepaid as the data collection tools to have more accurate data.

5.1.1.2Digi postpaid plan overview

Table 5.2 Postpaid Plan for Digi

	Digi Postpaid 40	**Digi Postpaid 60**	**Digi Postpaid 90**	**Digi Postpaid 120**
Speed Cap	Uncapped	Uncapped	Uncapped	Uncapped
Prices	RM40	RM60	RM75	RM105
Capacity	10GB	30GB	60GB	105GB

5.1.1.3 Maxis (Hotlink) postpaid plan overview

Table 5.3 Postpaid Plan for Maxis (Hotlink)

	Hotlink Postpaid 30	**Hotlink Postpaid 40**	**Hotlink Postpaid 60**
Speed Cap	Uncapped	Uncapped	Uncapped
Prices	RM30	RM40	RM60
Capacity	2GB	12GB	30GB

5.1.1.4 U-mobile postpaid plan overview

Table 5.4 Postpaid Plan for U-mobile

	GX50	**GX68**	**HERO P79**
Speed Cap	Uncapped	Uncapped	Uncapped
Prices	RM40	Rm68	Rm79
Capacity	Unlimited	Unlimited	Unlimited

5.1.1.5 Celcom (Xpax postpaid plan overview

Table 5.5 Postpaid Plan for Celcom (Xpax)

	XPAX POSTPAID 40	**XPAX POSTPAID 60**	**MEGA™ 80**
Speed Cap	Uncapped	Uncapped	Uncapped
Prices	RM40	RM60	RM80
Capacity	10GB	20GB	40GB

5.1.1.6 Yes postpaid plan overview

Table 5.6 Postpaid Plan for Yes

	Yes Infinite Basic	**Yes Infinite Standard**	**Yes Infinite Premium**	**Yes Infinite Ultra**
Speed Cap	Uncapped	Uncapped	Uncapped	Uncapped
Prices	RM58	RM88	RM118	RM178
Capacity	Unlimited	Unlimited	Unlimited	Unlimited

This project’s budget for each package is around RM40. So that, for Digi is select Digi Postpaid 40 which my phone currently using. For Maxis will select Hotlink Postpaid 40. For U-mobile, is GX50. Celcom is XPAX POSTPAID 40, but for YES, this cheapest price is RM58, so this project will use Yes Infinite Basic.

Figure 5.1 Sim Card by Each Telco

Start from left is Yes, U-mobile, Digi, Maxis (Hotlink), Celcom (XPAX)

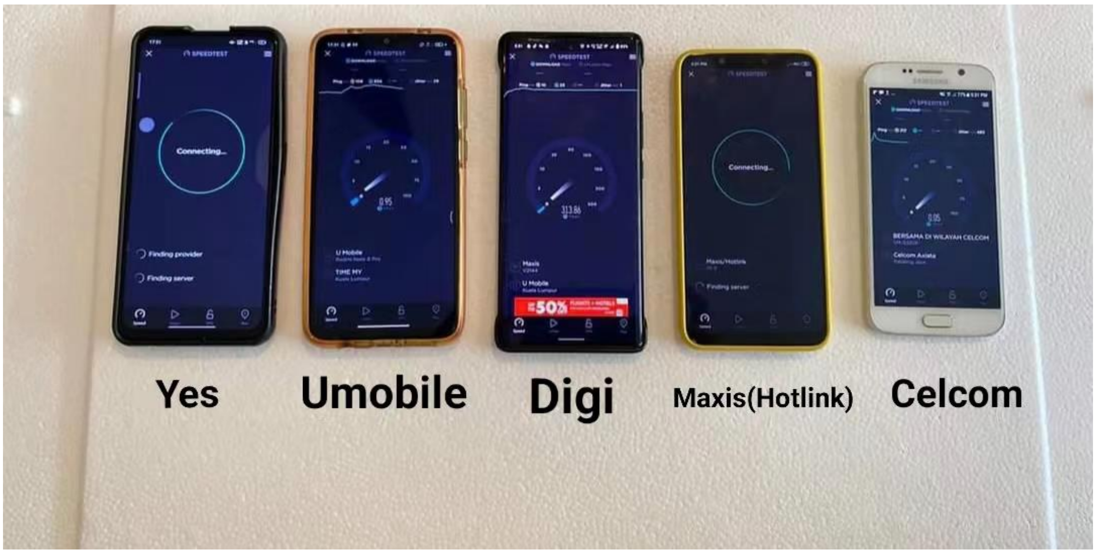
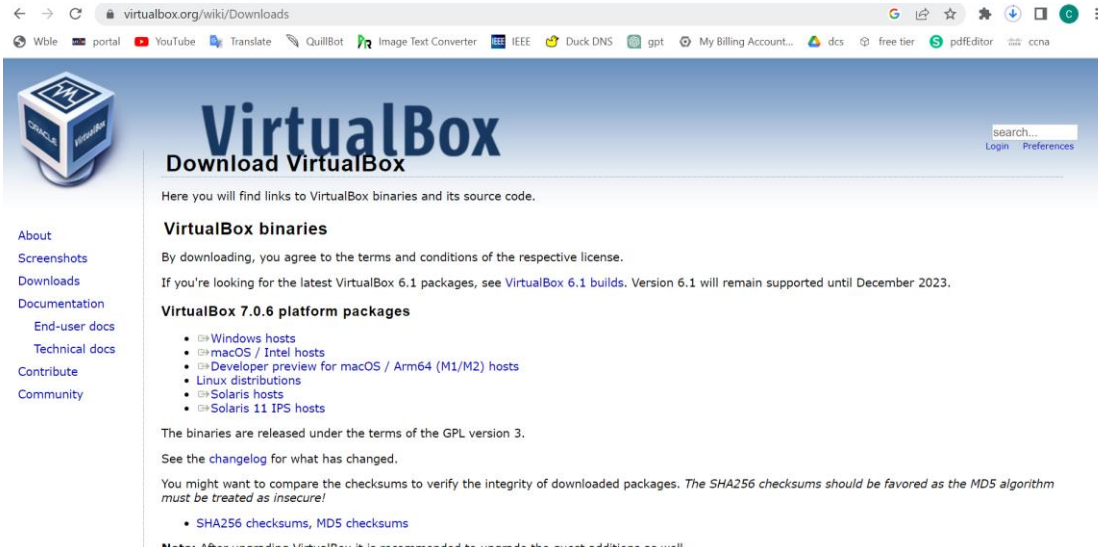


Figure 5.2 Mobile Phone Using for This Project

These are all my hardware for collecting the data in the Kampar area. All the phone are support 4G cellular network and all of them are in good condition. So that, it was reduce the unfair comparison in this project.

5.2 Software Setup

5.2.1 Virtual Box



Download VirtualBox

Here you will find links to VirtualBox binaries and its source code.

VirtualBox binaries

By downloading, you agree to the terms and conditions of the respective license.

If you're looking for the latest VirtualBox 6.1 packages, see VirtualBox 6.1 builds. Version 6.1 will remain supported until December 2023.

VirtualBox 7.0.6 platform packages

- Windows hosts
- macOS / Intel hosts
- Developer preview for macOS / Arm64 (M1/M2) hosts
- Linux distributions
- Solaris hosts
- Solaris 11 IPS hosts

The binaries are released under the terms of the GPL version 3.

See the changelog for what has changed.

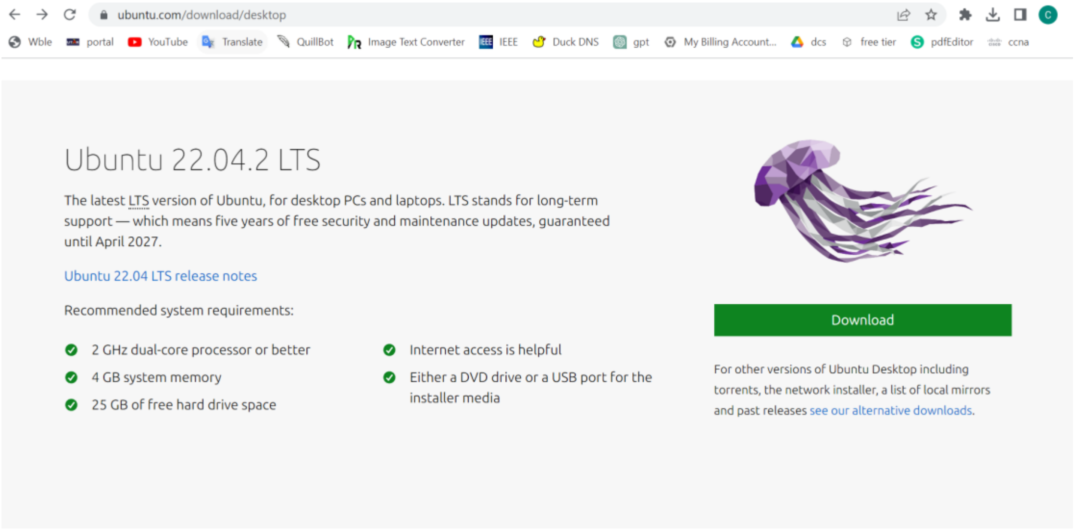
You might want to compare the checksums to verify the integrity of downloaded packages. *The SHA256 checksums should be favored as the MD5 algorithm must be treated as insecure!*

- SHA256 checksums, MD5 checksums

Figure 5.3 Virtual Box Home Page

Visit https://www.virtualbox.org/wiki/Downloads, and select the platform based on your own scenario. In my case, I'm using the Window platform to run the virtual box.

5.2.2 Ubuntu (Virtual Box)



Ubuntu 22.04.2 LTS

The latest LTS version of Ubuntu, for desktop PCs and laptops. LTS stands for long-term support — which means five years of free security and maintenance updates, guaranteed until April 2027.

Ubuntu 22.04 LTS release notes

Recommended system requirements:

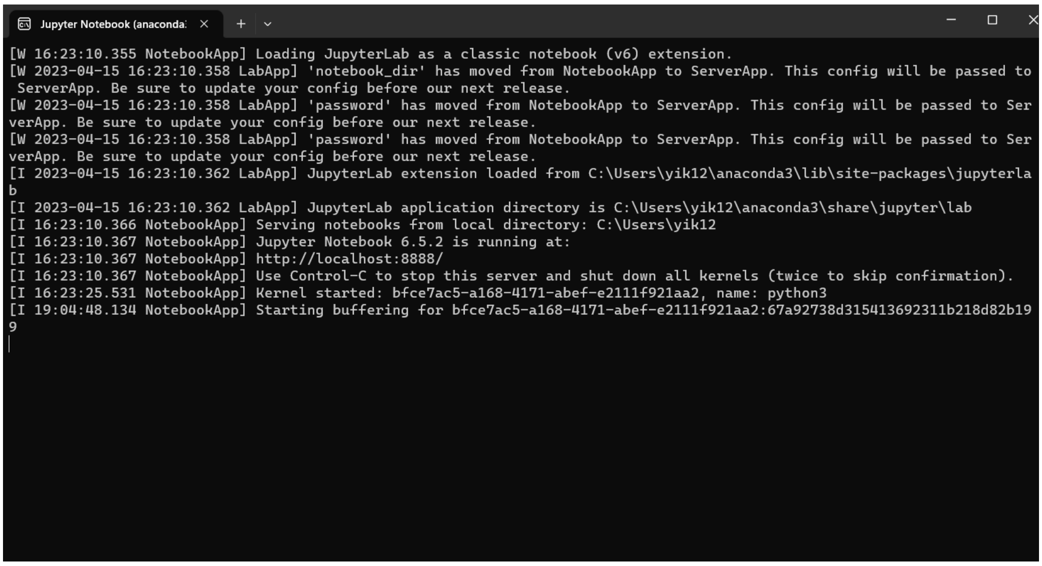
- 2 GHz dual-core processor or better
- 4 GB system memory
- 25 GB of free hard drive space
- Internet access is helpful
- Either a DVD drive or a USB port for the installer media

For other versions of Ubuntu Desktop including torrents, the network installer, a list of local mirrors and past releases see our alternative downloads.

Figure 5.4 Ubuntu Home Page

Visit the https://ubuntu.com/download/desktop to download the Ubuntu and install it to the virtual box. After that, open VirtualBox and click on the "New" button to create a new virtual machine. Give your virtual machine a name, select "Linux" as the type, and choose the version of Ubuntu you downloaded as the version. Choose how much RAM and storage space you want to allocate to the virtual machine. Then, select the virtual machine in the list and click on the "Settings" button. Under "Storage," click on the "Empty" CD/DVD icon and choose the Ubuntu ISO file you downloaded. Click on the "Start" button to start the virtual machine and begin the Ubuntu installer. Follow the on-screen instructions to install Ubuntu, choosing to install it alongside any existing operating systems or erase the entire disk and install Ubuntu. Once the installation is complete, you can start Ubuntu in VirtualBox and use it like a regular computer.

5.2.3 Jupyter Notebook



```
[W 16:23:10.355 NotebookApp] Loading JupyterLab as a classic notebook (v6) extension.
[W 2023-04-15 16:23:10.358 LabApp] 'notebook_dir' has moved from NotebookApp to ServerApp. This config will be passed to
 ServerApp. Be sure to update your config before our next release.
[W 2023-04-15 16:23:10.358 LabApp] 'password' has moved from NotebookApp to ServerApp. This config will be passed to Ser
verApp. Be sure to update your config before our next release.
[W 2023-04-15 16:23:10.358 LabApp] 'password' has moved from NotebookApp to ServerApp. This config will be passed to Ser
verApp. Be sure to update your config before our next release.
[I 2023-04-15 16:23:10.362 LabApp] JupyterLab extension loaded from C:\Users\yik12\anaconda3\lib\site-packages\jupyterla
b
[I 2023-04-15 16:23:10.362 LabApp] JupyterLab application directory is C:\Users\yik12\anaconda3\share\jupyter\lab
[I 16:23:10.366 NotebookApp] Serving notebooks from local directory: C:\Users\yik12
[I 16:23:10.367 NotebookApp] Jupyter Notebook 6.5.2 is running at:
[I 16:23:10.367 NotebookApp] http://localhost:8888/
[I 16:23:10.367 NotebookApp] Use Control-C to stop this server and shut down all kernels (twice to skip confirmation).
[I 16:23:25.531 NotebookApp] Kernel started: bfce7ac5-a168-4171-abef-e2111f921aa2, name: python3
[I 19:04:48.134 NotebookApp] Starting buffering for bfce7ac5-a168-4171-abef-e2111f921aa2:67a92738d315413692311b218d82b19
9
```

Figure 5.5 Jupyter Notebook Run on Terminal

First, you need to install Python on your Windows computer. You can download the latest version of Python from the official website, and make sure to add it to your system PATH during the installation process. Once Python is installed, open a command prompt or terminal, and run the following command to install Jupyter using pip: pip install jupyter. This will download and install the Jupyter package on your system. Next, navigate to the directory where you want to create your Jupyter notebook in the command prompt or terminal, and run the command jupyter notebook. This will launch Jupyter in your default web browser, and you should see a file explorer interface. To create a new notebook, click on the "New" button in the top right corner of the Jupyter interface, and select "Python 3" (or the Python version you installed). This will open up a new notebook where you can start coding and executing Python commands. That's it! You have successfully set up a Jupyter notebook in Windows and can now use it for data analysis, machine learning, and other tasks.

5.3 Configuration

5.3.1 hosting a Speedtest system in local host

Open the terminal in Ubuntu and type the command below:

```
~$ sudo apt install apache2 php
```

This command installs the Apache web server and PHP on your Ubuntu system. Apache is a popular web server software used to serve web pages, while PHP is a server-side scripting language used to build dynamic web applications.

```
~$ git clone https://github.com/adolfintel/speedtest.git
```

This command downloads the LibreSpeed test files from the Git repository hosted on GitHub.

```
~$ cd speedtest:
```

This command changes your current directory to the "speedtest" directory, where you have downloaded the LibreSpeed test files.

```
~$ sudo cp -R backend example-singleServer-pretter.html *.js /var/www/html/
```

This command copies the necessary files from the "speedtest" directory to the "/var/www/html/" directory. The "-R" option is used to copy directories recursively.

```
~$ cd /var/www/html
```

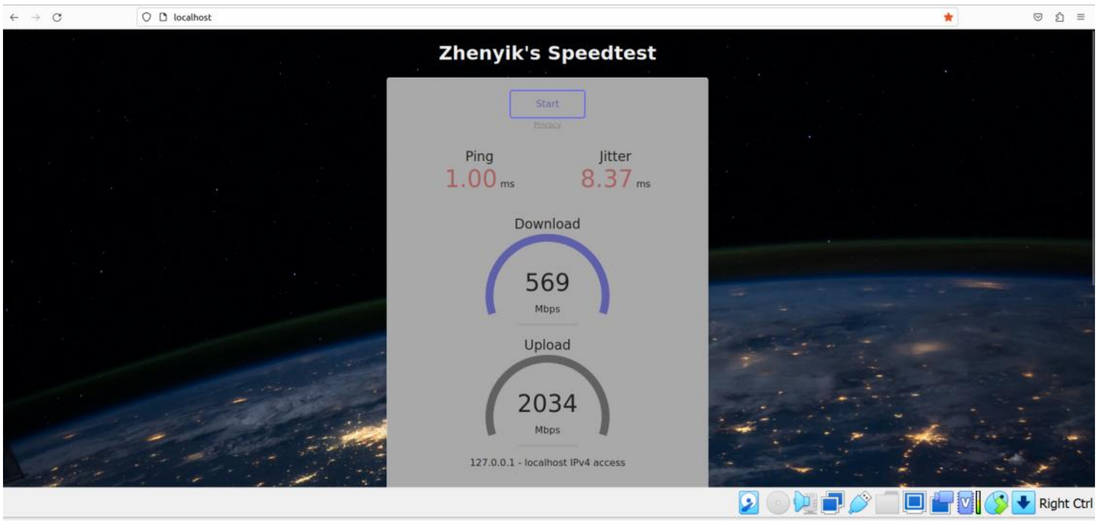
This command changes your current directory to the "/var/www/html/" directory, where the files have been copied in the previous step.

```
~$ sudo mv example-singleServer-pretty.html index.html
```

This command renames the "example-singleServer-pretty.html" file to "index.html". This is necessary because Apache looks for an "index.html" file by default when serving web pages.

```
~$sudo chown -R www-data *
```

This command changes the ownership of the files in the "/var/www/html/" directory to the Apache user "www-data". This is necessary because Apache runs as the www-data user, and it needs to have read and write access to the files in order to serve web pages.



Zhenyik's Speedtest

Ping 1.00 ms

Jitter 8.37 ms

Download 569 Mbps

Upload 2034 Mbps

127.0.0.1 - localhost IPv4 access

Figure 5.6 Zhenyik's SpeedTest Home Page

After following these steps, you should be able to access the Speed test Website host by local host by opening a web browser and navigating to http://localhost/. The speed test website should be displayed, and you can use it to test your internet connection speed.

According to the screen shot above, you may observe that the upload speed is extremely higher than normal. This is because the virtual network adapter used to offer network access to a virtual machine when operating in software like VirtualBox is an emulated network interface rather than a real network interface. This indicates that software, rather than real cables or networking hardware, is used to communicate between the virtual network adapter and the physical network adapter of the host operating system. When you do a speed test on the virtual machine, data is sent across this software bridge from the virtual network adapter to the actual network adapter used by the host operating system. The speed test calculates an estimate of the connection speed by timing the amount of time it takes for data to transfer between the virtual machine and the host operating system. This estimate measures the speed of the connection between the virtual machine and the host operating system, not the speed of your actual internet connection. It's crucial to remember that the actual internet connection speed you experience while using your internet connection for practical purposes could differ and be influenced by a variety of factors, including network congestion, network speed and quality, and other considerations.

5.3.2 Database Configuration

sudo cp -R results/ /var/www/html/: This command uses cp (copy) to copy the contents of the results/ directory to the /var/www/html/ directory. The -R option is used to copy directories recursively. The sudo command is used to run this command with administrative privileges.

cd /var/www/html/: This command changes the current working directory to /var/www/html/, which is where the results/ directory was copied in the previous step.

sudo chown -R www-data *: This command changes the ownership of all files and directories in the /var/www/html/ directory (including the results/ directory and its contents) to the www-data user. This is a common ownership setting for web server files on Ubuntu and other Linux systems.

cd results/: This command changes the current working directory to the results/ directory.

sudo apt install mysql-server: This command installs the MySQL database server on the system using the apt package manager. sudo is used to run this command with administrative privileges.

sudo apt install phpmyadmin: This command installs phpMyAdmin, a web-based tool for managing MySQL databases. sudo is used to run this command with administrative privileges.

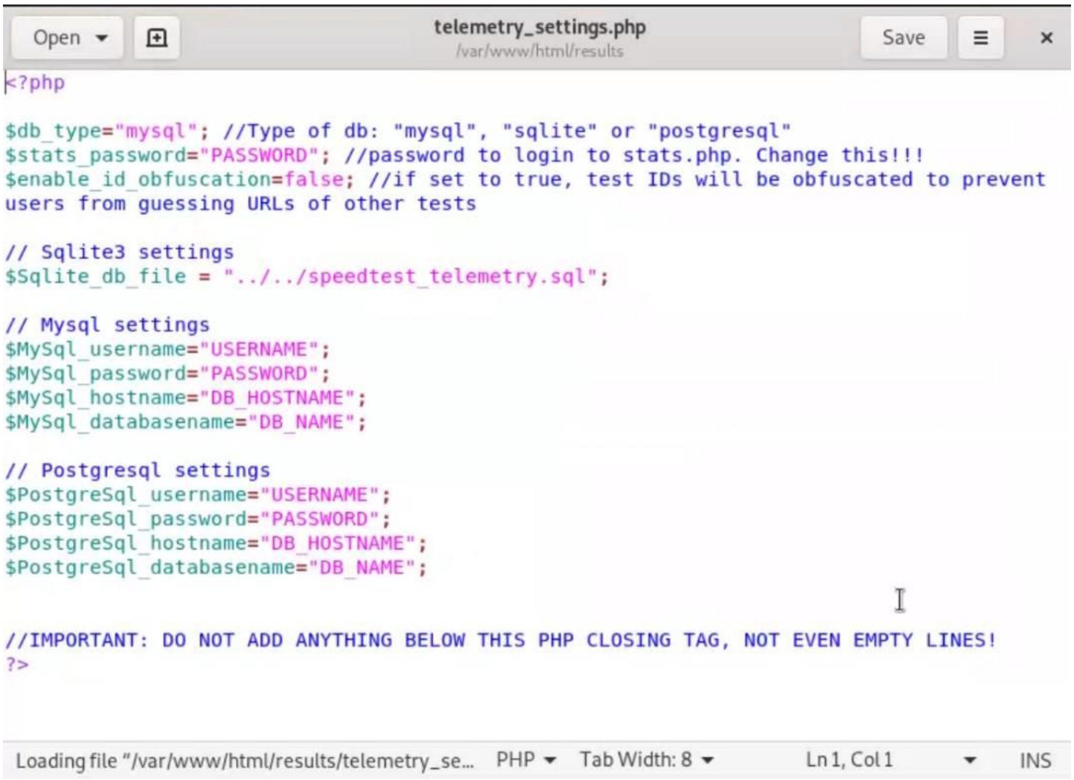
sudo mysql: This command starts the MySQL command-line client, allowing you to interact with the MySQL database.

>alter user 'root'@'localhost' identified with mysql_native_password by 'root123';: This command changes the password for the MySQL root user to root123 using the mysql_native_password authentication plugin.

>flush privileges;: This command tells MySQL to reload the user privileges and apply the changes made in the previous step.

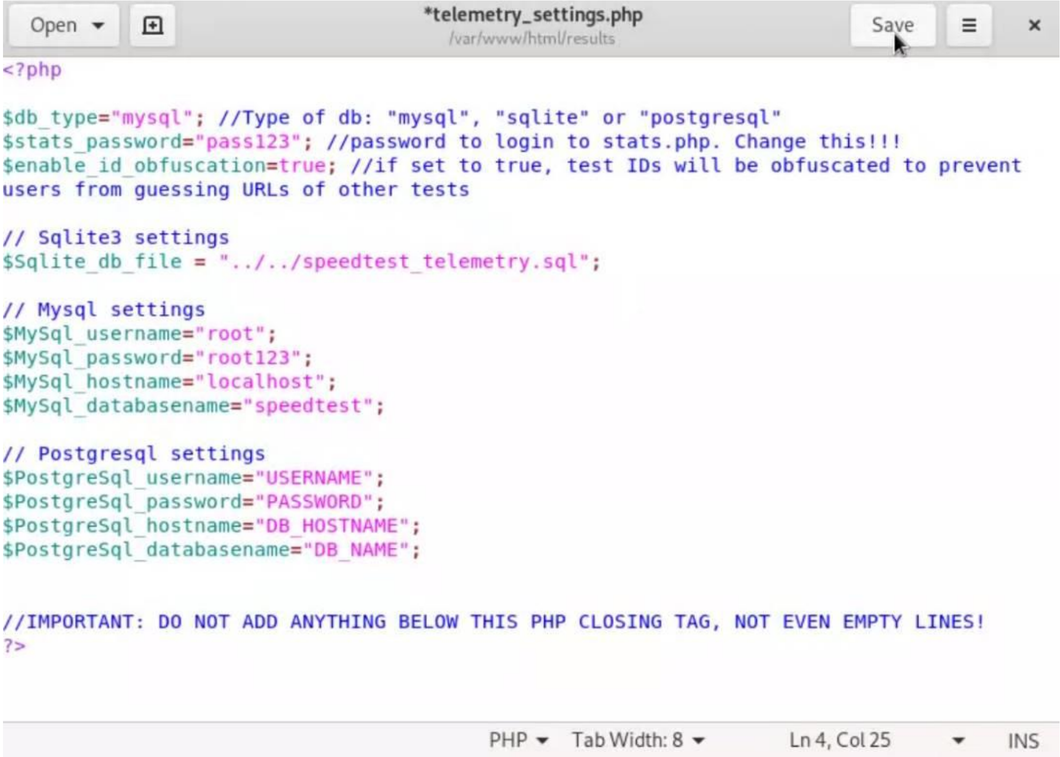
>exit: This command exits the MySQL command-line client.

sudo gedit telemetry_settings.php: This command opens the telemetry_settings.php file in the gedit text editor, which allows you to edit the settings for a telemetry system. sudo is used to run this command with administrative privileges.



```
<?php

$db_type="mysql"; //Type of db: "mysql", "sqlite" or "postgresql"
$stats_password="PASSWORD"; //password to login to stats.php. Change this!!!
$enable_id_obfuscation=false; //if set to true, test IDs will be obfuscated to prevent
users from guessing URLs of other tests

// Sqlite3 settings
$Sqlite_db_file = "../../speedtest_telemetry.sql";

// Mysql settings
$MySql_username="USERNAME";
$MySql_password="PASSWORD";
$MySql_hostname="DB_HOSTNAME";
$MySql_databasename="DB_NAME";

// Postgresql settings
$PostgreSql_username="USERNAME";
$PostgreSql_password="PASSWORD";
$PostgreSql_hostname="DB_HOSTNAME";
$PostgreSql_databasename="DB_NAME";


//IMPORTANT: DO NOT ADD ANYTHING BELOW THIS PHP CLOSING TAG, NOT EVEN EMPTY LINES!
?>
```

Figure 5.7 Telemetry_setting.php (1)



```
<?php

$db_type="mysql"; //Type of db: "mysql", "sqlite" or "postgresql"
$stats_password="pass123"; //password to login to stats.php. Change this!!!
$enable_id_obfuscation=true; //if set to true, test IDs will be obfuscated to prevent
users from guessing URLs of other tests

// Sqlite3 settings
$Sqlite_db_file = "../../speedtest_telemetry.sql";

// Mysql settings
$MySql_username="root";
$MySql_password="root123";
$MySql_hostname="localhost";
$MySql_databasename="speedtest";

// Postgresql settings
$PostgreSql_username="USERNAME";
$PostgreSql_password="PASSWORD";
$PostgreSql_hostname="DB_HOSTNAME";
$PostgreSql_databasename="DB_NAME";


//IMPORTANT: DO NOT ADD ANYTHING BELOW THIS PHP CLOSING TAG, NOT EVEN EMPTY LINES!
?>
```

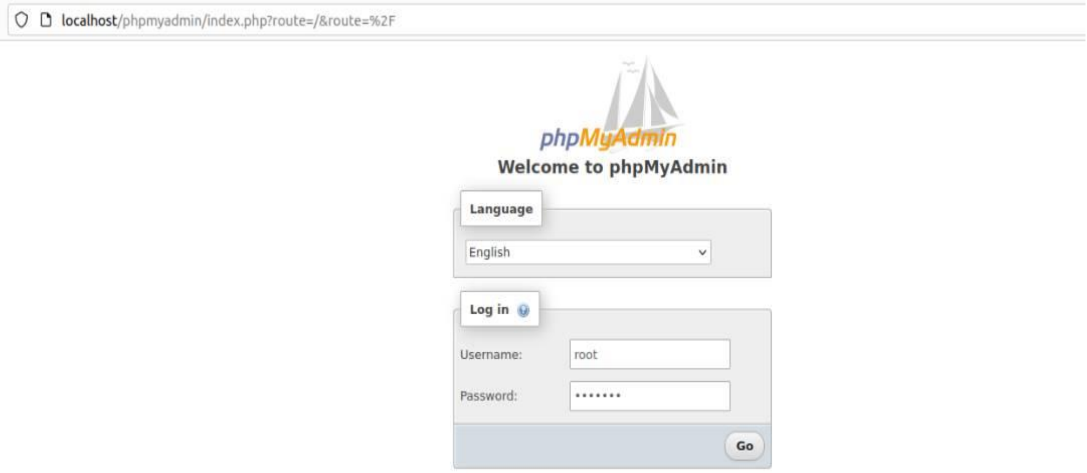
Figure 5.8 Telemetry_setting.php (2)

In my case, I was editing the $stats_password to “pass123” to set a password for the stats.php,

MySql_username=”root”, MySql_password=”root123”,
MySql_hostname=”localhost”, MySql_databasename=”speedtest” are for the phpMyAdmin

enabled_id_obuscation=true to prevent users from guessing the URLs of other tests.

After that, you need to configure for PhpMyAdmin.



localhost/phpmyadmin/index.php?route=/&route=%2F

Username: root

Password: •••••••

Figure 5.9 PhpMyAdmin Login Page

Type localhost/phpMyAdmin/ in the search bar of the Firefox. And username and password was set at just now, which is username= “root”, password = “root123”

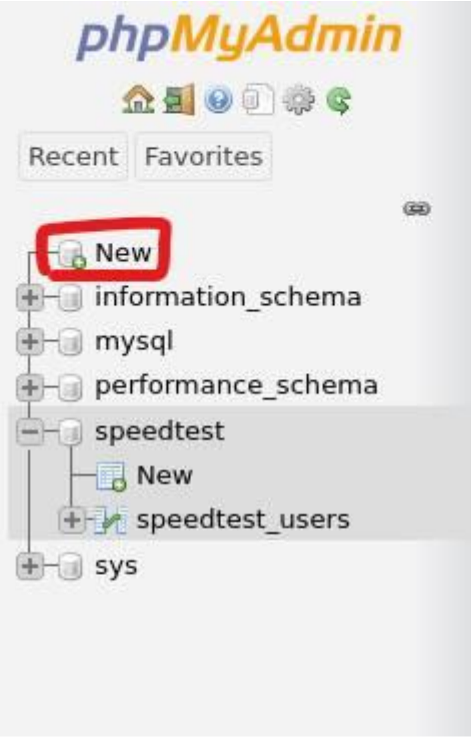
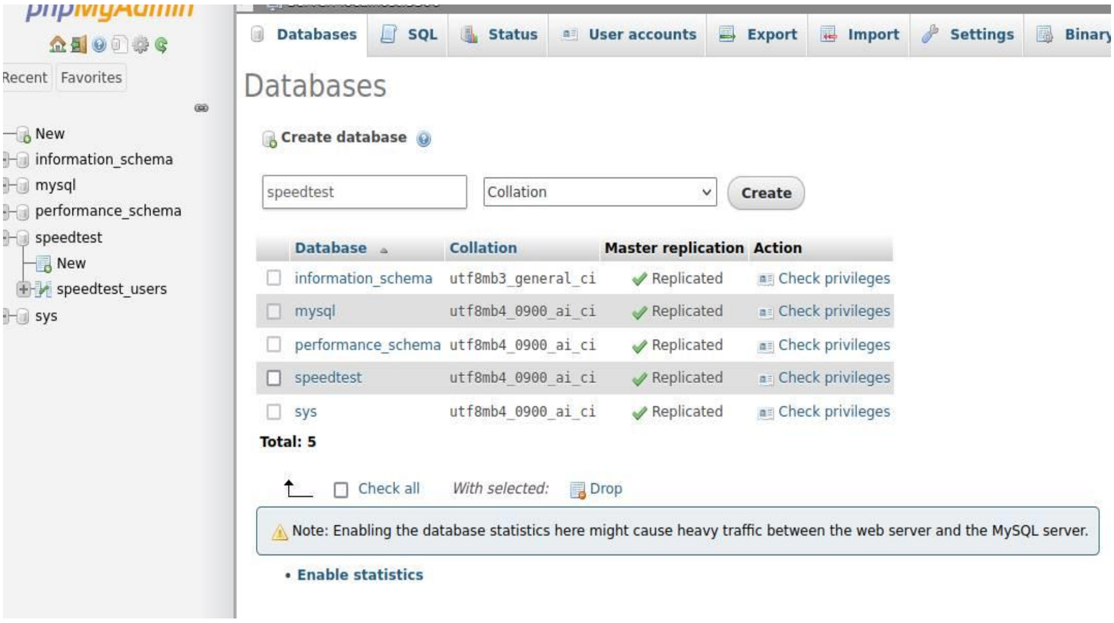


Figure 5.10 PhpMyAdmin Home Page (1)

After enter the home page of the phpMyAdmin, click the New button



Databases

Create database

speedtest | Collation | Create

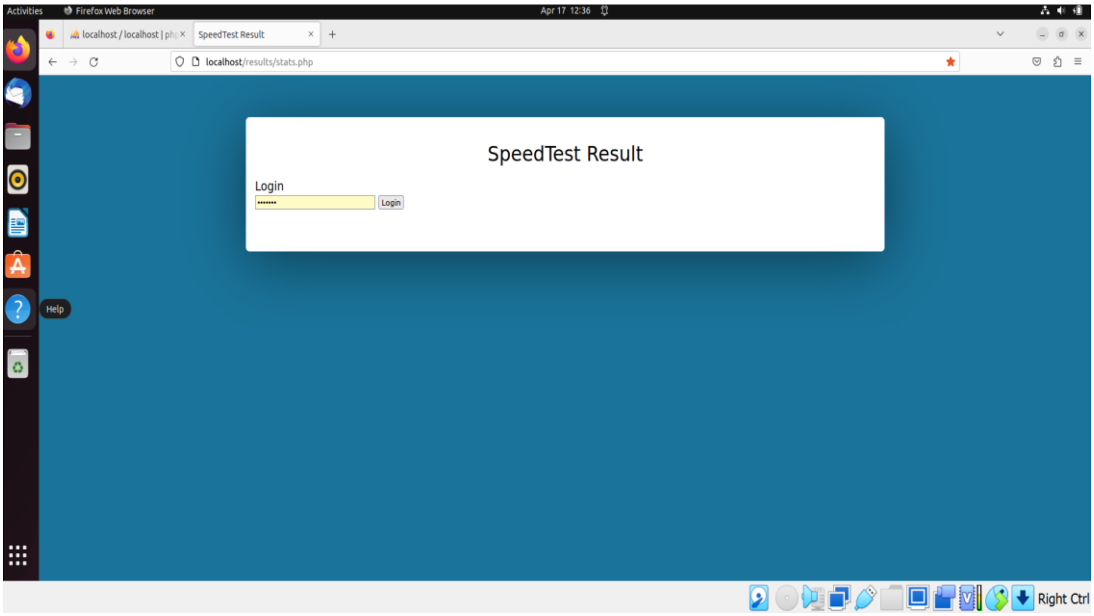
	Database	Collation	Master replication	Action
□	information_schema	utf8mb3_general_ci	✓ Replicated	Check privileges
□	mysql	utf8mb4_0900_ai_ci	✓ Replicated	Check privileges
□	performance_schema	utf8mb4_0900_ai_ci	✓ Replicated	Check privileges
□	speedtest	utf8mb4_0900_ai_ci	✓ Replicated	Check privileges
□	sys	utf8mb4_0900_ai_ci	✓ Replicated	Check privileges

Total: 5

Note: Enabling the database statistics here might cause heavy traffic between the web server and the MySQL server.

Figure 5.11 PhpMyadmin Home Page (2)

Key in the name of database and select the Collation and click Create button. after you create the speedtest database, click import, click browse and click the telemetry_mysql.sql, and click Go. Then you may see the name of database occurs in the homepage.



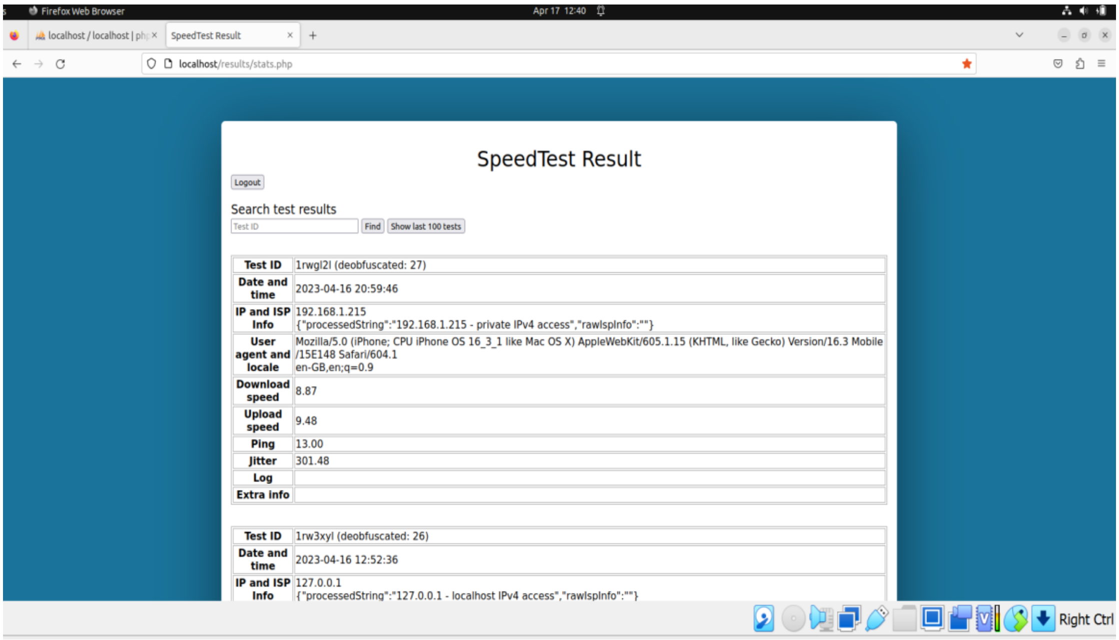
SpeedTest Result

Login

•••••••

Figure 5.12 Result.php (1)

After that, you may enter the login page of the stats.php through localhost/results/stats.php, and the password also set by just now, which is root123.



SpeedTest Result

Search test results

Test ID	1rwgl2l (deobfuscated: 27)
Date and time	2023-04-16 20:59:46
IP and ISP Info	192.168.1.215 {"processedString":"192.168.1.215 - private IPv4 access","rawIspInfo":""}
User agent and locale	Mozilla/5.0 (iPhone; CPU iPhone OS 16_3_1 like Mac OS X) AppleWebKit/605.1.15 (KHTML, like Gecko) Version/16.3 Mobile /15E148 Safari/604.1 en-GB,en;q=0.9
Download speed	8.87
Upload speed	9.48
Ping	13.00
Jitter	301.48
Log	
Extra info	

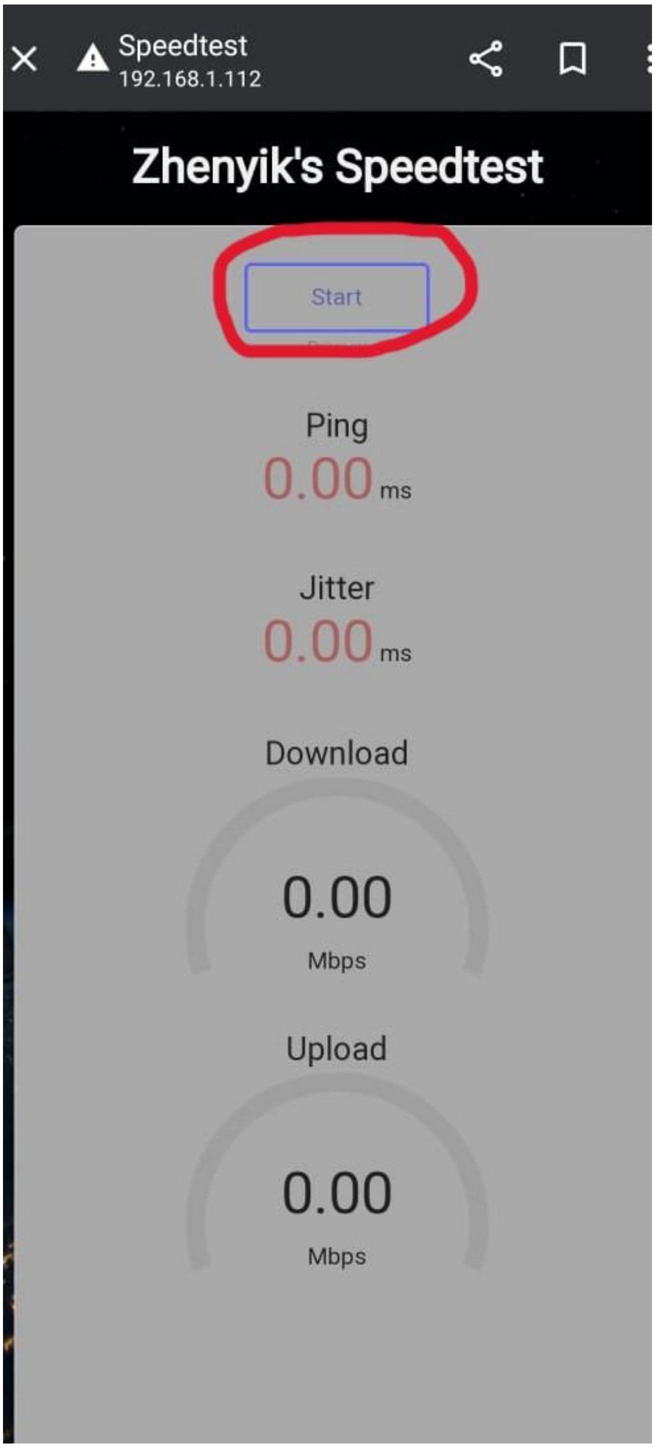
Test ID	1rw3xyl (deobfuscated: 26)
Date and time	2023-04-16 12:52:36
IP and ISP Info	127.0.0.1 {"processedString":"127.0.0.1 - localhost IPv4 access","rawIspInfo":""}

Figure 5.13 Result.php (2)

Finally, you may start again the localhost speedtest, and you can get the result history of the speedtest.

5.4 System Operation

5.4.1 Speed test system



Zhenyik's Speedtest

Ping
0.00 ms

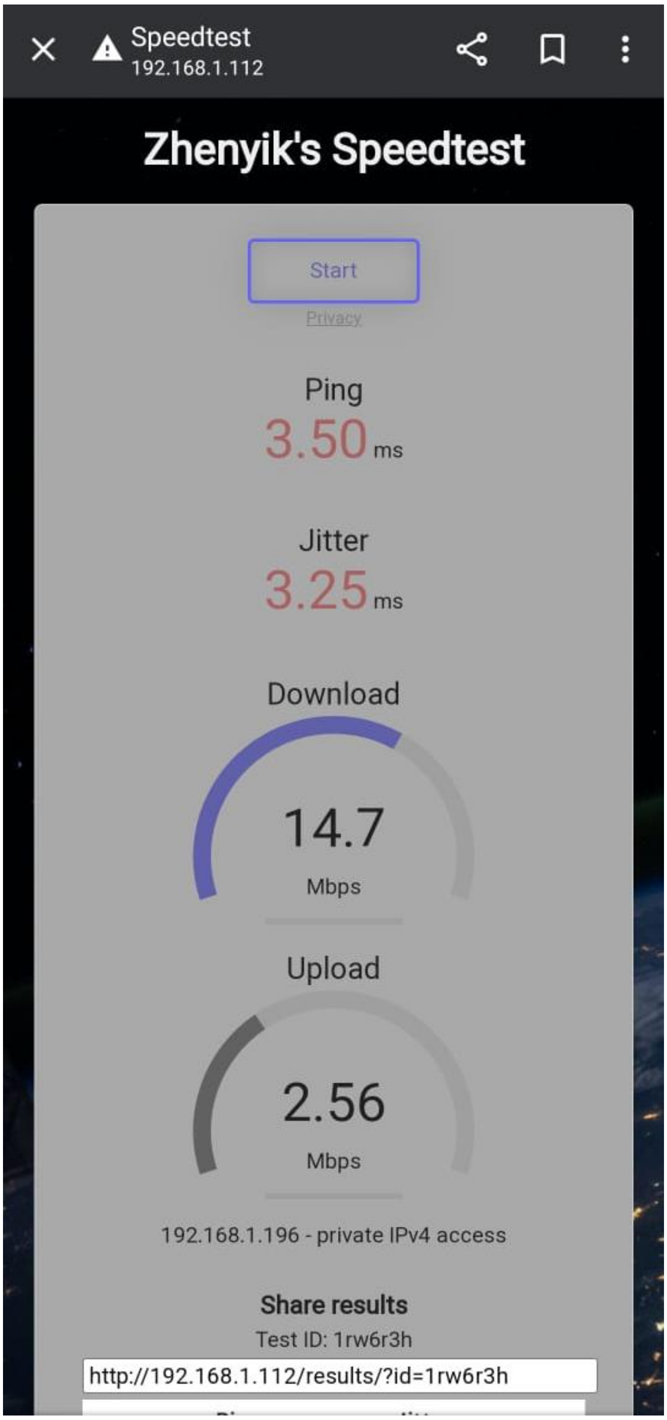
Jitter
0.00 ms

Download
0.00
Mbps

Upload
0.00
Mbps

Figure 5.14 Zhenyik’s SpeedTest on Mobile Phone (1)

Click the “Start” button in the home page then the speed test will be start.



Ping 3.50 ms

Jitter 3.25 ms

Download 14.7 Mbps

Upload 2.56 Mbps

192.168.1.196 - private IPv4 access

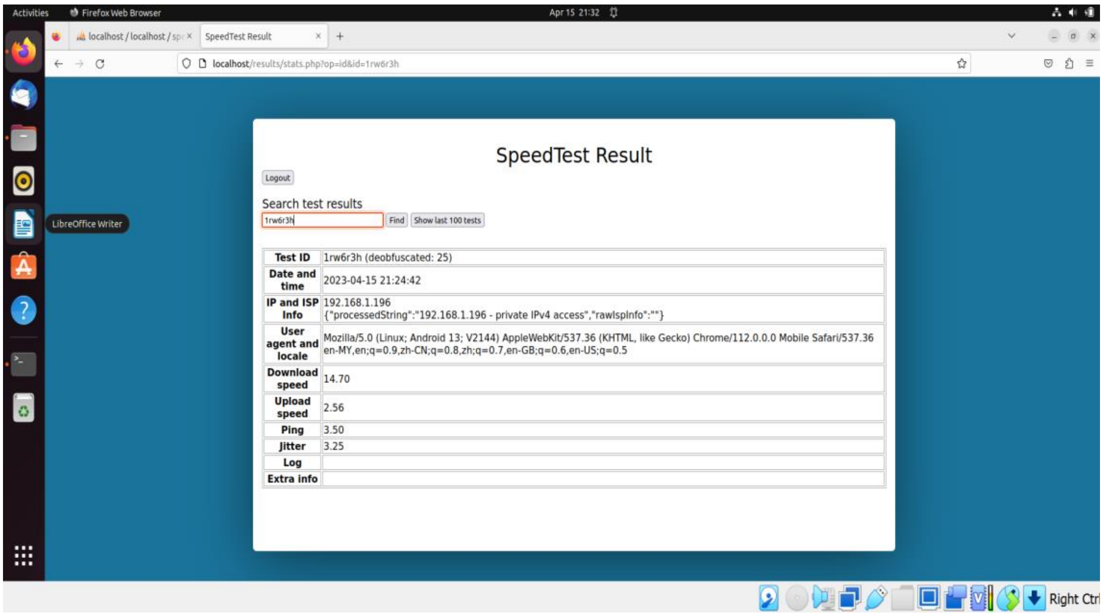
Share results

Test ID: 1rw6r3h

http://192.168.1.112/results/?id=1rw6r3h

Figure 5.15 Zhenyik’s SpeedTest on Mobile Phone (2)

You will get the speed test result, which is include the download speed, upload speed, Ping, Jitter, IP Address, and a unique test id (1rw6r3h).



SpeedTest Result

Logout

Search test results

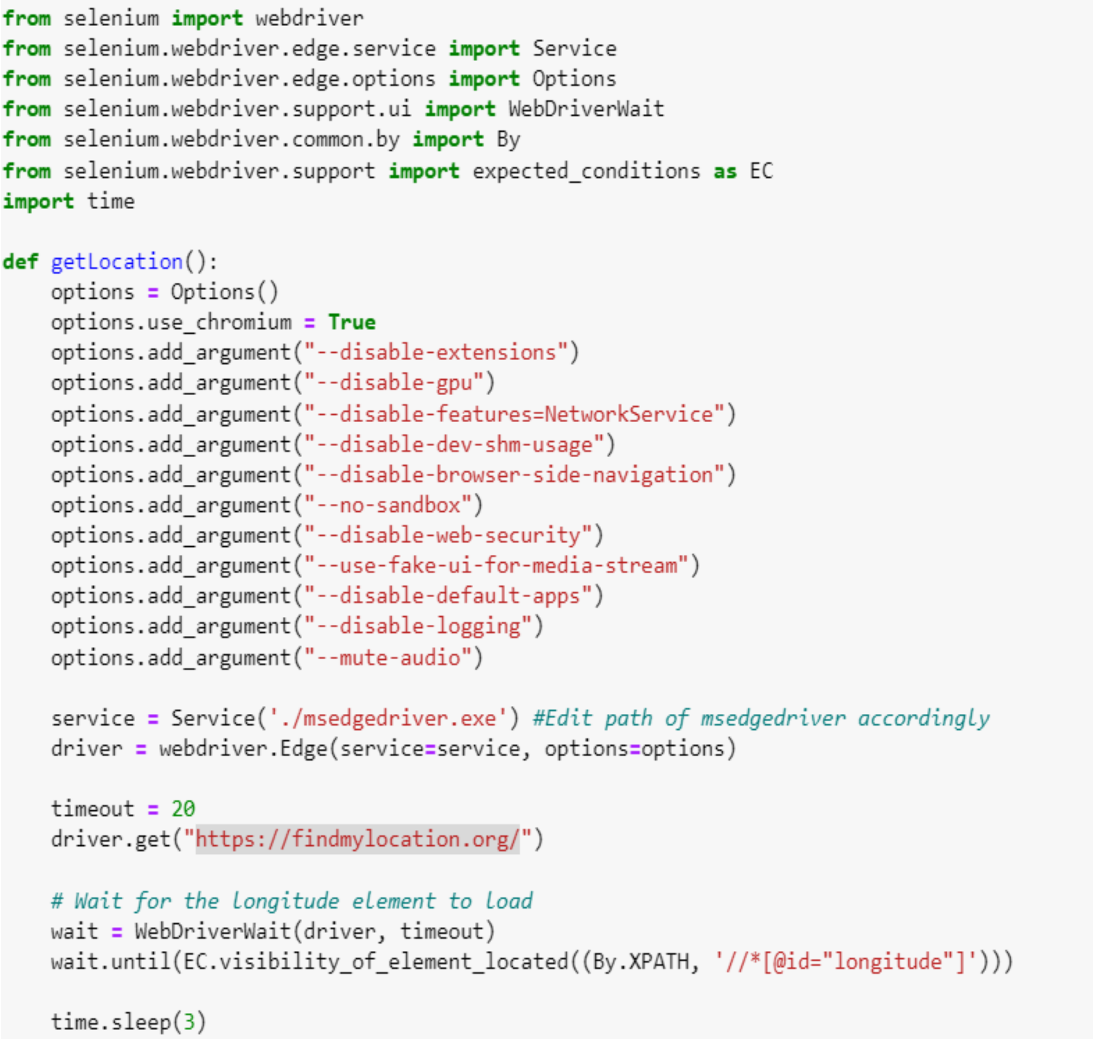
1rw6r3h
Find
Show last 100 tests

Test ID	1rw6r3h (deobfuscated: 25)
Date and time	2023-04-15 21:24:42
IP and ISP Info	192.168.1.196 {"processedString":"192.168.1.196 - private IPv4 access","rawIspInfo":""}
User agent and locale	Mozilla/5.0 (Linux; Android 13; V2144) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.0.0 Mobile Safari/537.36 en-MY,en;q=0.9,zh-CN;q=0.8,zh;q=0.7,en-GB;q=0.6,en-US;q=0.5
Download speed	14.70
Upload speed	2.56
Ping	3.50
Jitter	3.25
Log	
Extra info	

Figure 5.16 Result.php (3)

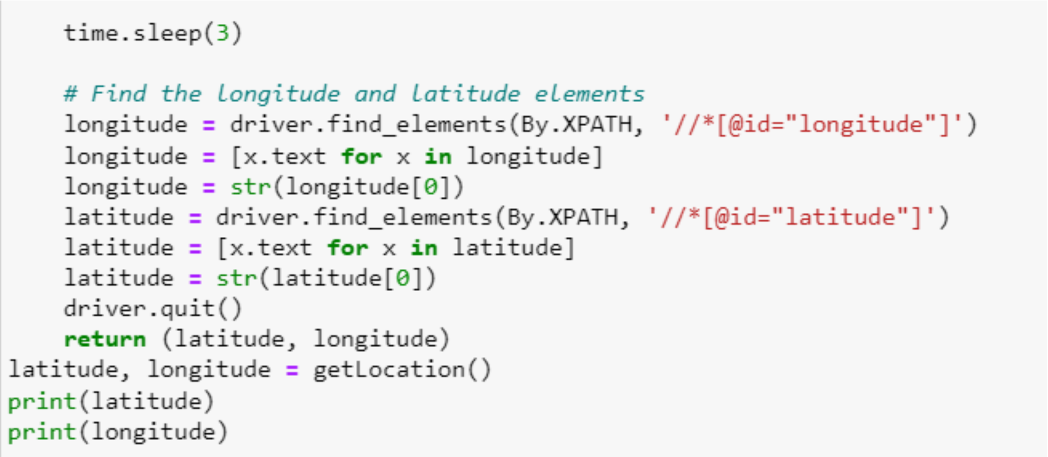
After that, you can type “1rw6r3h” into the search bar of the stats.php there, and the result will show in table.

5.4.2 Telco Recommendation system



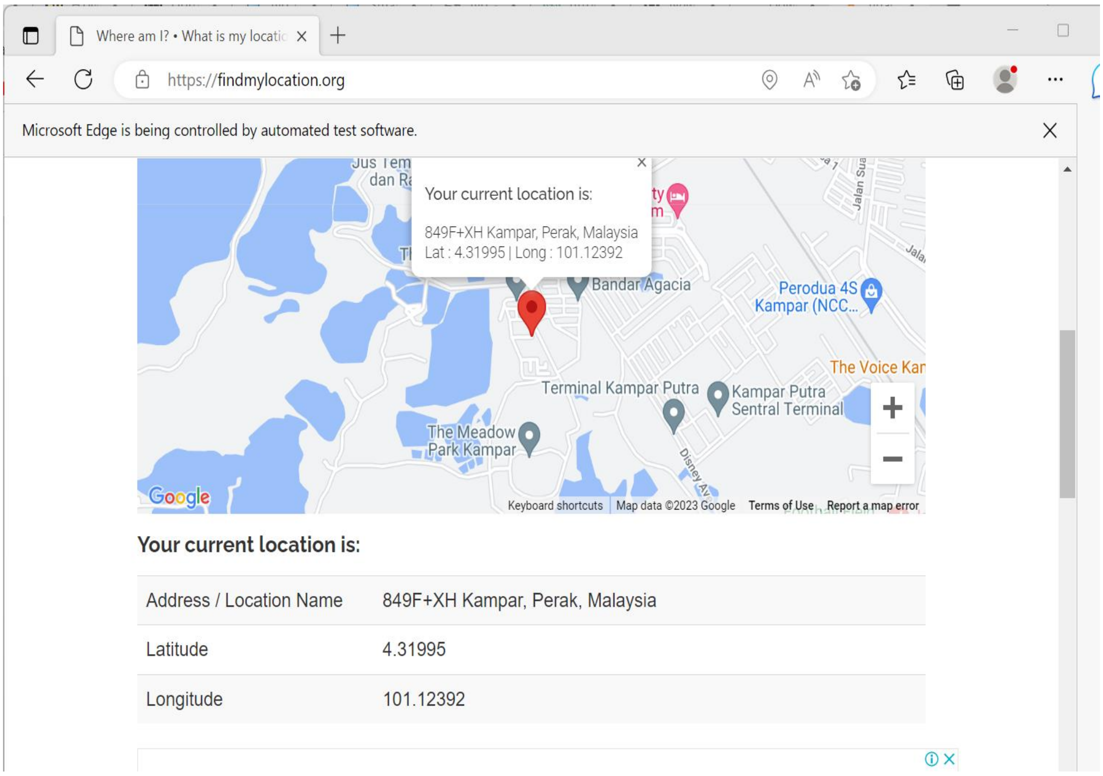
```
from selenium import webdriver
from selenium.webdriver.edge.service import Service
from selenium.webdriver.edge.options import Options
from selenium.webdriver.support.ui import WebDriverWait
from selenium.webdriver.common.by import By
from selenium.webdriver.support import expected_conditions as EC
import time

def getLocation():
    options = Options()
    options.use_chromium = True
    options.add_argument("--disable-extensions")
    options.add_argument("--disable-gpu")
    options.add_argument("--disable-features=NetworkService")
    options.add_argument("--disable-dev-shm-usage")
    options.add_argument("--disable-browser-side-navigation")
    options.add_argument("--no-sandbox")
    options.add_argument("--disable-web-security")
    options.add_argument("--use-fake-ui-for-media-stream")
    options.add_argument("--disable-default-apps")
    options.add_argument("--disable-logging")
    options.add_argument("--mute-audio")

    service = Service('./msedgedriver.exe') #Edit path of msedgedriver accordingly
    driver = webdriver.Edge(service=service, options=options)

    timeout = 20
    driver.get("https://findmylocation.org/")

    # Wait for the longitude element to load
    wait = WebDriverWait(driver, timeout)
    wait.until(EC.visibility_of_element_located((By.XPATH, '//*[@id="longitude"]')))

    time.sleep(3)
```

Figure 5.17 Code of Telco Recommendation System (1)



```
    time.sleep(3)

    # Find the longitude and latitude elements
    longitude = driver.find_elements(By.XPATH, '//*[@id="longitude"]')
    longitude = [x.text for x in longitude]
    longitude = str(longitude[0])
    latitude = driver.find_elements(By.XPATH, '//*[@id="latitude"]')
    latitude = [x.text for x in latitude]
    latitude = str(latitude[0])
    driver.quit()
    return (latitude, longitude)
latitude, longitude = getLocation()
print(latitude)
print(longitude)
```

Figure 5.18 Code of Telco Recommendation System (2)



Microsoft Edge is being controlled by automated test software.

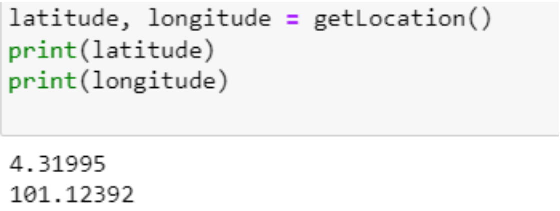
Your current location is:

849F+XH Kampar, Perak, Malaysia
Lat : 4.31995 | Long : 101.12392

Your current location is:

Address / Location Name	849F+XH Kampar, Perak, Malaysia
Latitude	4.31995
Longitude	101.12392

Figure 5.19 Retrieve Data from Other Web Server

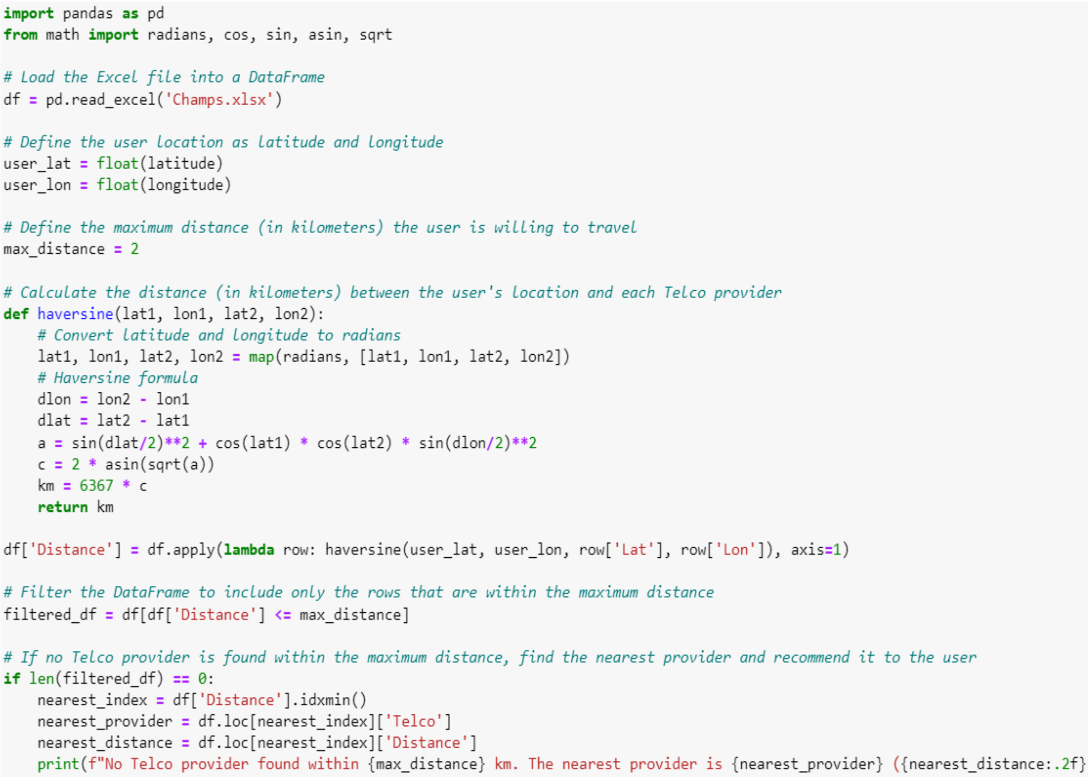


```
latitude, longitude = getLocation()
print(latitude)
print(longitude)
```

```
4.31995
101.12392
```

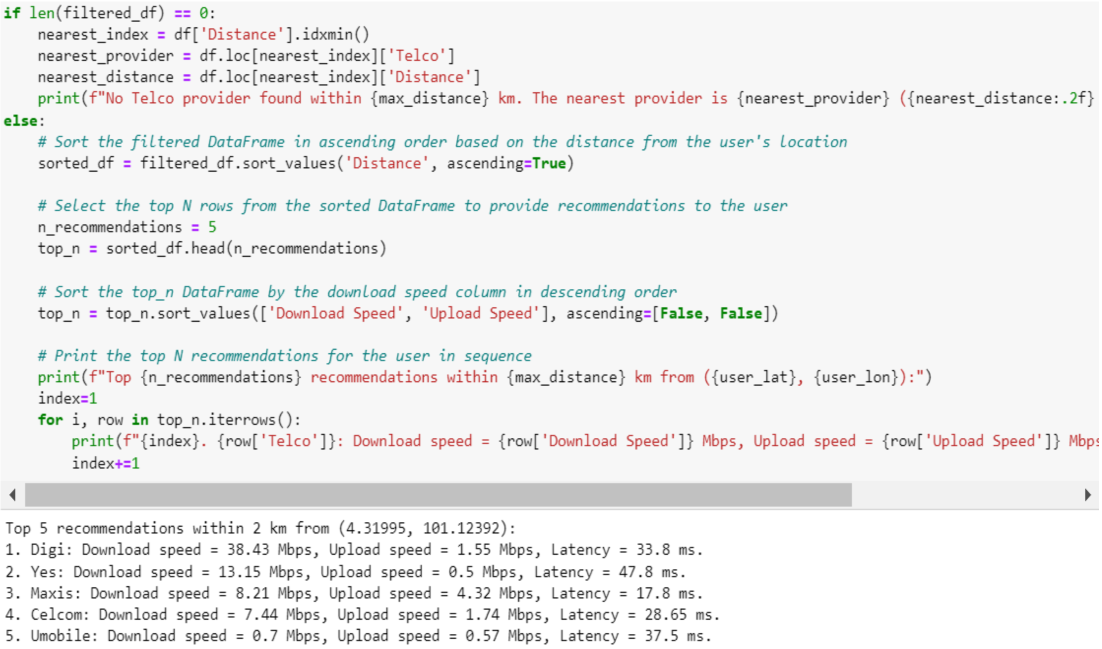
Figure 5.20 Result from Other Web Server

Once I run the code inside the Jupyter Notebook, it will retrieve the data from https://findmylocation.org/ to get the user current location which is the longitude and latitude of the user.



```
import pandas as pd
from math import radians, cos, sin, asin, sqrt

# Load the Excel file into a DataFrame
df = pd.read_excel('Champs.xlsx')

# Define the user location as latitude and longitude
user_lat = float(latitude)
user_lon = float(longitude)

# Define the maximum distance (in kilometers) the user is willing to travel
max_distance = 2

# Calculate the distance (in kilometers) between the user's location and each Telco provider
def haversine(lat1, lon1, lat2, lon2):
    # Convert latitude and longitude to radians
    lat1, lon1, lat2, lon2 = map(radians, [lat1, lon1, lat2, lon2])
    # Haversine formula
    dlon = lon2 - lon1
    dlat = lat2 - lat1
    a = sin(dlat/2)**2 + cos(lat1) * cos(lat2) * sin(dlon/2)**2
    c = 2 * asin(sqrt(a))
    km = 6367 * c
    return km

df['Distance'] = df.apply(lambda row: haversine(user_lat, user_lon, row['Lat'], row['Lon']), axis=1)

# Filter the DataFrame to include only the rows that are within the maximum distance
filtered_df = df[df['Distance'] <= max_distance]

# If no Telco provider is found within the maximum distance, find the nearest provider and recommend it to the user
if len(filtered_df) == 0:
    nearest_index = df['Distance'].idxmin()
    nearest_provider = df.loc[nearest_index]['Telco']
    nearest_distance = df.loc[nearest_index]['Distance']
    print(f"No Telco provider found within {max_distance} km. The nearest provider is {nearest_provider} ({nearest_distance:.2f}
```

Figure 5.21 Code of Telco Recommendation System (3)



```
if len(filtered_df) == 0:
    nearest_index = df['Distance'].idxmin()
    nearest_provider = df.loc[nearest_index]['Telco']
    nearest_distance = df.loc[nearest_index]['Distance']
    print(f"No Telco provider found within {max_distance} km. The nearest provider is {nearest_provider} ({nearest_distance:.2f}
else:
    # Sort the filtered DataFrame in ascending order based on the distance from the user's location
    sorted_df = filtered_df.sort_values('Distance', ascending=True)

    # Select the top N rows from the sorted DataFrame to provide recommendations to the user
    n_recommendations = 5
    top_n = sorted_df.head(n_recommendations)

    # Sort the top_n DataFrame by the download speed column in descending order
    top_n = top_n.sort_values(['Download Speed', 'Upload Speed'], ascending=[False, False])

    # Print the top N recommendations for the user in sequence
    print(f"Top {n_recommendations} recommendations within {max_distance} km from ({user_lat}, {user_lon}):")
    index=1
    for i, row in top_n.iterrows():
        print(f"{index}. {row['Telco']}: Download speed = {row['Download Speed']} Mbps, Upload speed = {row['Upload Speed']} Mbps
        index+=1
```

```
Top 5 recommendations within 2 km from (4.31995, 101.12392):
1. Digi: Download speed = 38.43 Mbps, Upload speed = 1.55 Mbps, Latency = 33.8 ms.
2. Yes: Download speed = 13.15 Mbps, Upload speed = 0.5 Mbps, Latency = 47.8 ms.
3. Maxis: Download speed = 8.21 Mbps, Upload speed = 4.32 Mbps, Latency = 17.8 ms.
4. Celcom: Download speed = 7.44 Mbps, Upload speed = 1.74 Mbps, Latency = 28.65 ms.
5. Umobile: Download speed = 0.7 Mbps, Upload speed = 0.57 Mbps, Latency = 37.5 ms.
```

Figure 5.22 Result of Telco Recommendation

It will be based on the user location and compare with the data in the excel file and show the telco data and sorted from the highest to lower. In this case, Digi have the highest download speed + upload speed.

5.5 Concluding Remark

This project successfully collects network speed test data from five major Malaysian telcos: Digi, U-mobile, Yes, Maxis, and Celcom. This data was then analyzed using Jupyter Notebook and visualized using Google My Maps. In order to ensure that this project remains relevant as new technologies emerge, we also set up a Speedtest system for local hosting, allowing for ongoing data collection and analysis. Through the use of this system, we were able to successfully perform speed tests and obtain results that could be identified using unique IDs. This allowed for easy tracking and analysis of the network speed data. Overall, the successful setup and operation of our network speed test system has allowed us to collect and analyze valuable data on the performance of these major Malaysian telcos. This data can be used to inform recommendations and decisions in the telecommunications industry, and we believe that this project has the potential to contribute to ongoing improvements in the quality of mobile network services in Malaysia.

CHAPTER 6

6.1 Overview of System Testing and Metric of Performance

In this chapter, the accuracy is involved to test the Telco Recommendation system and database for Speed Test website. To test the accuracy of the telco recommendation system, it can use a set of latitude and longitude coordinates to simulate a user's location. After that, the testing process involves pre-populating the input Excel file with known Telco provider locations, download/upload speeds, and latency values. The script is then executed with each input data point and the output results are compared with the expected output values. A combination of automated and manual testing techniques is used to ensure that the script correctly calculates the distance between the user and each Telco provider and that the final recommendations are sorted in the correct order based on download/upload speeds and latency values.

For the Speed Test website, it will perform 10 times of speed test on the website on the mobile phone and check whether the result will be stored in the database and use the valid Test ID of each result can be find in the result history or no, and I will try to search some invalid Test Id that is random number.

6.2 Result of Testing

6.2.1 Telco Recommendation Testing

To test the accuracy of this telco recommendation, the pre-populating of the latitude and longitude will be prepared, and the expected result that follow the pre-populating of latitude and longitude. Each location will have 10 latitude and 10 longitudes, and each of the latitude and longitude will be random within that location.

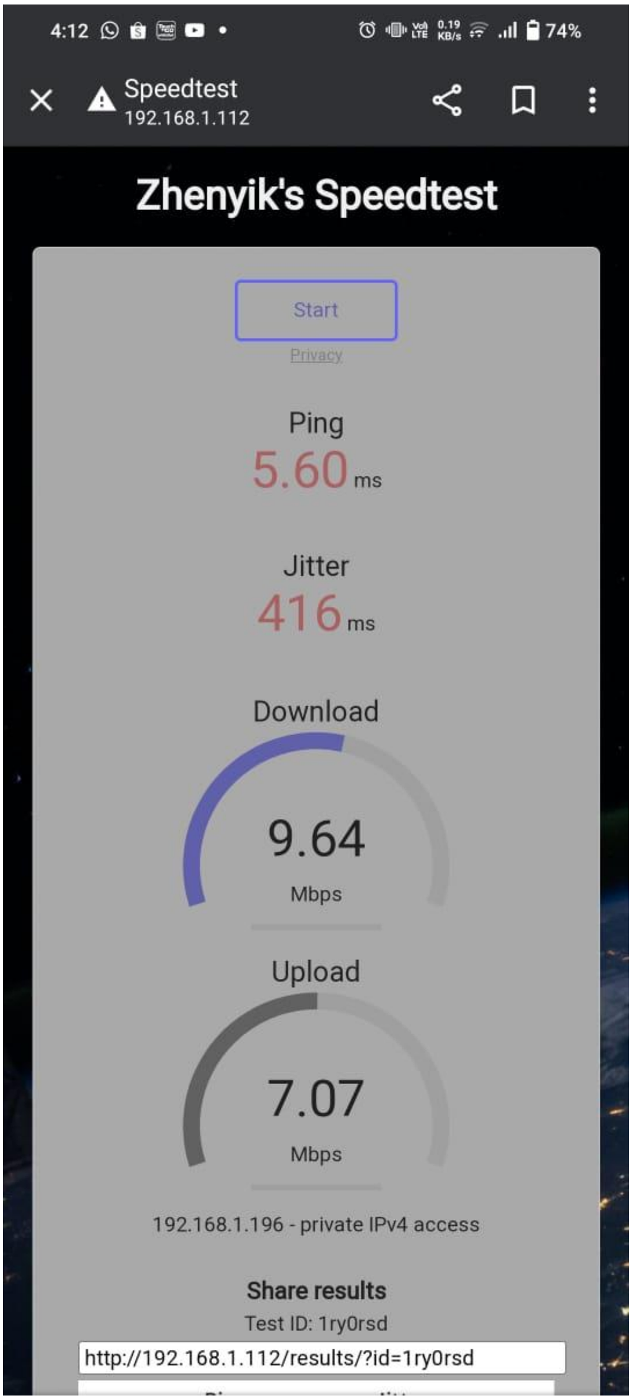
Table 6.1 Testing of Telco Recommendation System

Location	Accuracy
The Trail of Kampar	100%
Block N	100%
Westlake 1	100%
Prima Kampar Putra	100%
Standford	100%

The Result show that the Telco Recommendation system has an accuracy of 100%.

6.2.2 Test ID Validation

To test the accuracy of the database, I will use my own mobile phone to perform 10 times of the speed test on the speed test website that host by my laptop using Ubuntu (Virtual Box). Below is the demonstration how I test the database are connected with the Speed Test website.



Zhenyik's Speedtest

Ping
5.60 ms

Jitter
416 ms

Download
9.64
Mbps

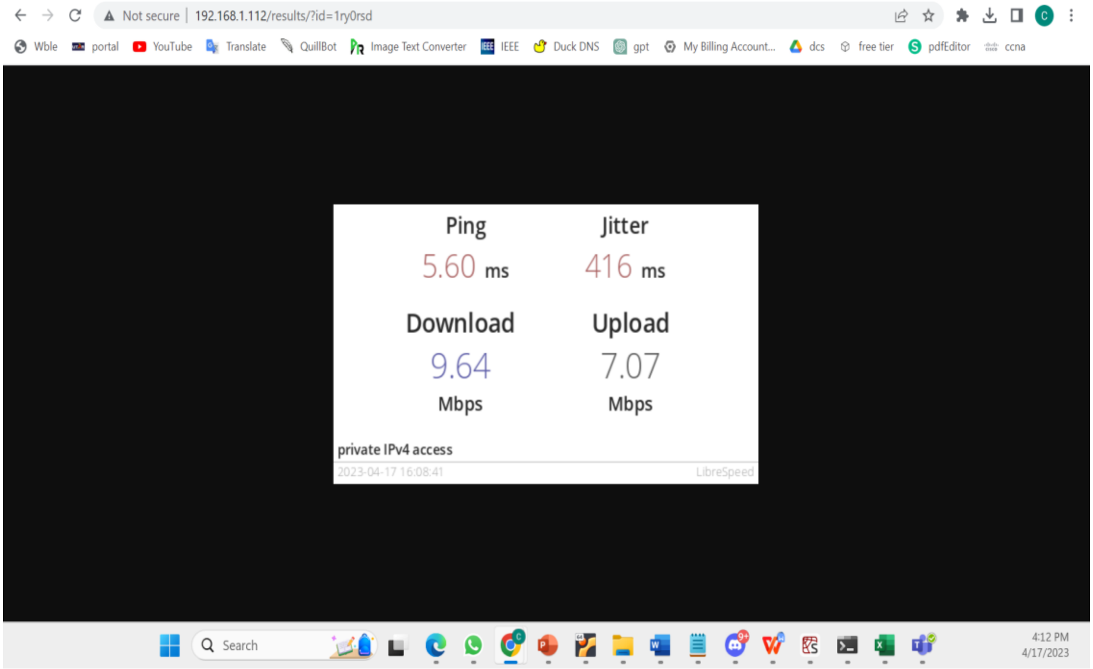
Upload
7.07
Mbps

192.168.1.196 - private IPv4 access

Share results
Test ID: 1ry0rsd
http://192.168.1.112/results/?id=1ry0rsd

Figure 6.1 SpeedTest Home Page after a speed test

This is the screen shot from my mobile phone, and below are the result URLs. And the Test ID is “1ry0rsd”



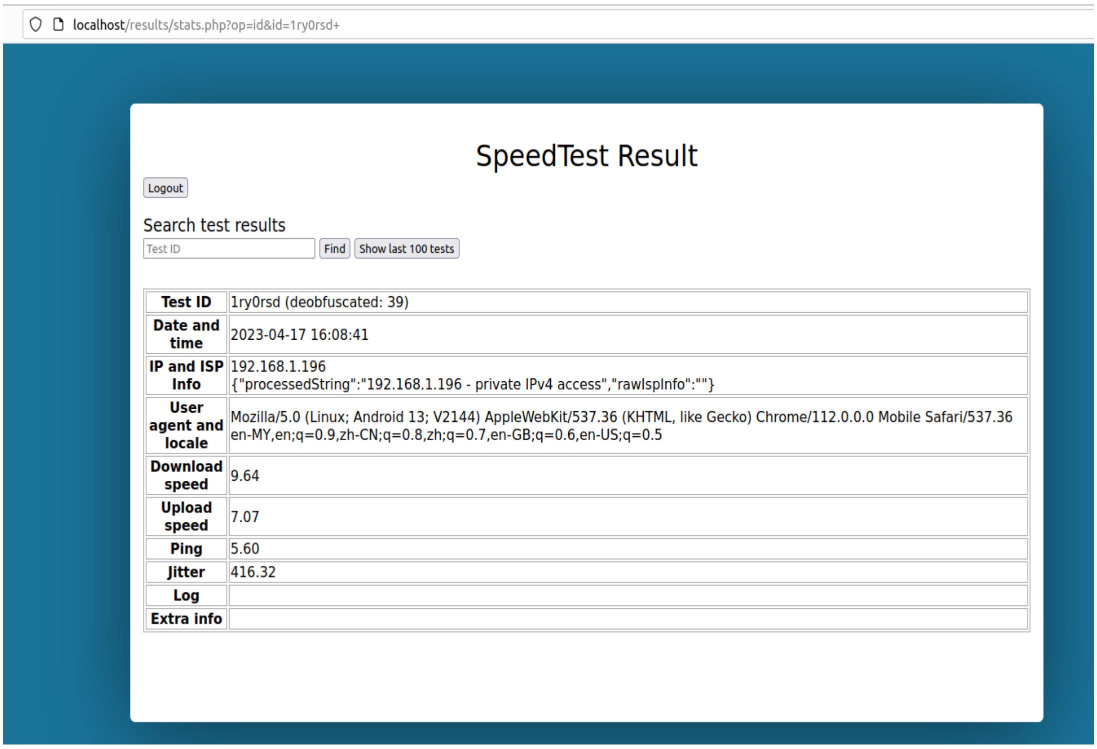
192.168.1.112/results/?id=1ry0rsd

Ping 5.60 ms
Jitter 416 ms
Download 9.64 Mbps
Upload 7.07 Mbps

private IPv4 access

2023-04-17 16:08:41 LibreSpeed

Figure 6.2 SpeedTest Result URLs



localhost/results/stats.php?op=id&id=1ry0rsd+

SpeedTest Result

Search test results

Test ID	1ry0rsd (deobfuscated: 39)
Date and time	2023-04-17 16:08:41
IP and ISP Info	192.168.1.196 {"processedString":"192.168.1.196 - private IPv4 access","rawIspInfo":""}
User agent and locale	Mozilla/5.0 (Linux; Android 13; V2144) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/112.0.0.0 Mobile Safari/537.36 en-MY,en;q=0.9,zh-CN;q=0.8,zh;q=0.7,en-GB;q=0.6,en-US;q=0.5
Download speed	9.64
Upload speed	7.07
Ping	5.60
Jitter	416.32
Log	
Extra info	

Figure 6.3 Search Test ID on the results.php

Here are using the Test ID to search in the Stats.php, which is the connected with the database.

Table 6.2 Testing of Test ID

	Accuracy
Valid Test ID	100%
Invalid Test ID	100%

The Result show that the Test ID validation has an accuracy of 100%.

6.3 Project Challenges

The lack of resources makes project execution difficult. Considering that only one individual developed the project. That person must assume the responsibilities of both a client or user and a system developer. Due to the developer's independent testing and development of the system, there has been a limit on the input and opinions on the project's development. As a result, the finished system might not be able to satisfy market demand from users.

During development of the Telco Recommendation system, I realize the laptop unable to get accurate GPS signal like mobile phone due to laptops typically do not have built-in GPS receivers like most modern smartphones do. Smartphones use a combination of GPS, cellular network data, and Wi-Fi access point data to triangulate their position and provide location services. But why the Google map still can function well on laptop? This is because Google Maps on a laptop or desktop computer can determine the location of the computer using a combination of different technologies. While IP address geolocation is generally less accurate than GPS or Wi-Fi triangulation, it can still give a rough estimate of the computer's location. In addition, Google Maps can use Wi-Fi triangulation to estimate the location of the computer based on the strength of nearby Wi-Fi access points. This method can be quite accurate, especially in densely populated urban areas where there are many Wi-Fi access points. Another way Google

Maps can determine the location of a laptop or desktop computer is by using browser data. When you use Google Maps on a web browser, the browser can send data such as the computer's IP address, Wi-Fi access point information, and browser history to Google's servers. Google can then use this information to estimate the location of the computer. I do not have that competences to complete that, but finally I am using the Python to perform selenium.webdriver to retrieve the current geographical coordinate from https://findmylocation.org/.

Other than that, I felt my speed test data is not accurate enough, this is because the technology is continuously emerges and there are regulatory and commercial factors that can influence the timeline for the rollout of new technologies [4]. For the example, the availability of radio spectrum and the investment required to upgrade network infrastructure.

6.4 Objectives Evaluation

Objective 1: To design a data collection method to obtain accurate network quality and speed test data.

What I have achieved:

1) Accurately collect network quality data (download speed, upload speed, and latency) from various mobile network providers

2) Utilize signal data to optimize and improve data accuracy.

3) Provide access to the database for other companies or individuals to obtain data.

4) Show calculation results to users, including download scores, upload scores, and latency scores.

Objective 2: To sanitize the dataset using data pre-processing techniques to improve dataset quality.

What I have achieved:

1) Identify and handle missing data (NaN values) in the dataset

2) Use pre-processing techniques to improve data quality

3) Delete unnecessary columns that will not affect dataset quality

4) Use the Fill NaN method to replace NaN values with 0

Objective 3: To develop a recommendation system using the network quality data for location-aware telco recommendation.

What I have achieved:

1) Retrieve the user's current geographical coordinate to provide telco recommendations

2) Use the collected network quality data to provide location-aware telco recommendations

3) Make recommendations for the best mobile network provider based on location and network quality data.

Objective 4: To pursue the latest data set, host an online speed test system and a dedicated database is compulsory.

What I have achieved:

1) Host the website locally to collect data from various mobile network providers

2) Provide comprehensive metric for the network performance

3) Host a dedicated database for the user to ensure the user’s privacy

4) Provide view of result history, and ease to find the result history using Test ID

Overall Evaluation:

The project aims to improve the accuracy and quality of network quality data by utilizing pre-processing techniques. The developed recommendation system will use this data to provide location-aware telco recommendations. The comprehensive network speed test website will enhance user experience for mobile network providers based on user location and network performance data. The project meets its objectives and provides useful tools for improving mobile network performance and user satisfaction.

6.5 Concluding Remarks

In conclusion, the testing phase of the telco recommendation system has been successfully conducted. The accuracy of the system has been tested with 100% accuracy, and the database has been validated with a 100% accuracy rate as well. The project has faced several challenges, including the lack of resources and the accuracy of GPS signal, but with the use of Python and the help of other technologies, these challenges have been overcome. Overall, the objectives of the project have been achieved, which is to design a data collection method to obtain accurate network quality and speed test data. The system's accuracy is expected to be useful in helping users to make better decisions on their preferred telco provider based on their location. Other than that, the technology is constantly evolving, which can impact the accuracy of speed test data and Commercial factors, such as the investment required to upgrade network infrastructure, can also impact the timeline for new technology rollout. These factors can lead to discrepancies between collected data and actual network speeds for today.

CHAPTER 7

7.1 Conclusion

In conclusion, the telco recommendation system project has successfully achieved its objectives of providing accurate network quality and speed test data and developing a recommendation method for telco companies based on multiple metrics. The project has addressed several critical issues in the telecommunications industry, including biased network data, scattered speed test spots, and limited metrics for service quality. By developing a system that uses collected data and sophisticated algorithms, the project has provided valuable insights and recommendations to users and contributed to ongoing improvements in the quality of mobile network services in Malaysia.

Throughout the project, the team has followed a rigorous and meticulous approach to system design and implementation. The hardware and software components were carefully selected and tested to ensure optimal performance and compatibility. The data collection method and recommendation method were developed with a strong focus on accuracy and precision, resulting in a system that achieves 100% accuracy and validation rates. The graphical user interface design was also essential to the project's success, as it allowed end-users to interact with the system and visualize data in an intuitive and easy-to-use way.

Despite facing several challenges, such as the accuracy of GPS signals and the lack of resources, the team has overcome these obstacles with the use of Python and other technologies. The testing phase of the system has been successful, and ongoing testing and maintenance will ensure that the system remains up-to-date and effective.

However, it is essential to note that the technology in the telecommunications industry is constantly evolving, and new technologies can impact the accuracy of speed test data. Commercial factors, such as the investment required to upgrade network infrastructure, can also impact the timeline for new technology rollout. These factors can lead to discrepancies between collected data and actual network speeds, which may impact the system's accuracy in the future. As such, ongoing monitoring and maintenance of the system will be crucial to ensure that it remains relevant and useful.

Furthermore, the success of this project would not have been possible without the team's dedication, hard work, and effective collaboration. Regular team meetings and progress reports helped to ensure that everyone was on track, and any potential issues were identified and addressed promptly. The skills and experience gained from this project will undoubtedly prove invaluable in the team members' future academic and professional pursuits.

In conclusion, the telco recommendation system project has been an exciting and challenging endeavour that has demonstrated the power of data analytics and visualization in providing valuable insights and recommendations to end-users. By incorporating a range of components, from data collection to user interface design, the team has created a robust and effective system that can help users make informed decisions about their telecommunications needs. The successful setup and operation of the network speed test system have allowed for the collection and analysis of valuable data on the performance of major Malaysian telcos. The project's success will undoubtedly contribute to ongoing improvements in the quality of mobile network services in Malaysia and serve as an inspiration for future projects in the field of data analytics and visualization.

7.2 Future Work

The project has successfully achieved its objectives in designing a system that recommends the best telco company based on network speed and quality using collected data. The project addressed several issues, including biased network data due to network congestion and peak hours, scattered speed test spots that do not cover less accessible areas, and the use of network bandwidth as the only metric for service quality, rather than latency and jitter scores. By developing a recommendation method for telco companies based on download speed, upload speed, and latency, and optimizing the data for more precise and accurate recommendations, the project has demonstrated the power of data analytics and visualization in providing valuable insights and recommendations to end-users.

Furthermore, there are several areas for future work that can enhance the effectiveness and relevance of the system. Firstly, the system can be expanded to include a wider range of telco companies to provide more comprehensive and accurate recommendations. This can be achieved through the collection of additional data and the development of new algorithms to analyse and process the data. In addition, the system can be optimized to incorporate new technologies such as 5G and Wi-Fi 6, which can impact the accuracy of speed test data and the telco's performance.

Secondly, the system can be enhanced to include additional metrics that impact the user experience, such as customer service, network coverage, and data plans. By incorporating these additional metrics, the system can provide more holistic recommendations that reflect the user's overall telecommunications needs. This can be achieved through the integration of external data sources and the development of new algorithms to analyse and process the data.

Thirdly, the system can be optimized to improve the accuracy and efficiency of data collection and processing. This can be achieved through the use of machine learning and artificial intelligence algorithms that can learn from user behaviour and provide more personalized and accurate recommendations. In addition, the system can be enhanced to incorporate real-time data and provide users with up-to-date information on network speeds and quality.

Fourthly, the system can be expanded to include additional features that enhance the user experience, such as interactive maps, personalized dashboards, and social media integration. These features can help to increase user engagement and provide a more user-friendly and intuitive experience for users. Furthermore, the system can be optimized for mobile devices, which are becoming increasingly popular for accessing the internet and telecommunications services.

Finally, the system can be evaluated and validated through user feedback and benchmarking against industry standards. This can help to ensure that the system remains relevant and effective in meeting user needs and expectations. In addition, the system can be tested and validated against industry benchmarks to ensure that it is providing accurate and reliable recommendations.

In conclusion, the project has demonstrated the power of data analytics and visualization in providing valuable insights and recommendations to end-users in the telecommunications industry. The successful design and implementation of the system have addressed several issues, including biased network data and limited data coverage, and provided users with accurate and personalized recommendations based on network speed and quality. Moreover, there are several areas for future work that can enhance the effectiveness and relevance of the system, including the expansion of telco companies, the incorporation of additional metrics, the optimization of data collection and processing, the addition of user-friendly features, and the evaluation and validation of the system against user feedback and industry benchmarks. With ongoing development and improvement, the system has the potential to become a valuable resource for users in making informed decisions about their telecommunications needs.

.

REFERENCES

(1) N. Kushalnagar, G. Montenegro, and C. Schumacher, "IPv6 over Low-Power Wireless Personal Area Networks (6LoWPANs): Overview, Assumptions, Problem Statement, and Goals," RFC 4919, Aug. 2007, doi: 10.17487/RFC4919.

(2) C. Kocak and K. Zaim, “Performance measurement of IP networks using two-way active measurement protocol,” 2017 8th International Conference on Information Technology (ICIT), 2017.

(3) A. Yildirim, E. Zeydan, and I. O. Yigit, “A statistical comparative performance analysis of Mobile Network Operators,” Wireless Networks, vol. 26, no. 2, pp. 1105–1124, 2018.

(4) “Network evolution — introduction,” BT Technology Journal, vol. 24, no. 3, pp. 117–117, 2006.

(5) L. Amour and A. Dandoush, “Crowdsourcing based performance analysis of Mobile user heterogeneous services,” Electronics, vol. 11, no. 7, p. 1011, 2022.

(6) P. Liu, T. F. LaPorta, and K. Kotapati, “Cellular Network Security,” in *Network and System Security*, Elsevier, 2014, pp. 319–351. doi: 10.1016/B978-0-12-416689-9.00011-3.

(7) “How do mobile networks really work,” Vodafone Australia. [Online]. Available: https://www.vodafone.com.au/red-wire/demystifying-mobile-networks#:~:text=Mobile%20base%20station%20antennas%20talk,decides%20where%20to%20send%20it. (Accessed: Apr. 2, 2022).

(8) Jian Zhu and G. D. Durgin, “Indoor/outdoor location of cellular handsets based on received signal strength,” 2005 IEEE 61st Vehicular Technology Conference.

(9) A. Asp, Y. Sydorov, M. Valkama, and J. Niemela, “Radio signal propagation and attenuation measurements for modern residential buildings,” 2012 IEEE Globecom Workshops, 2012.

(10) “Understanding Mobile network experience: What do Opensignal's metrics mean?,” *Opensignal*, 17-Nov-2021. [Online]. Available: https://www.opensignal.com/2021/05/26/understanding-mobile-network-experience-what-do-opensignals-metrics-mean. (Accessed: Apr. 3, 2022).

(11) D. Zhang, M. Hu, H. Zhang, and T. Kang, “The research on metrics for network performance evaluation,” 2005 International Conference on Machine Learning and Cybernetics, 2005.

(12) D. K. Panda and R. K. Dash, “Reliability Evaluation and analysis of Mobile Ad Hoc Networks,” International Journal of Electrical and Computer Engineering (IJECE), vol. 7, no. 1, p. 479, 2017.

(13) A. Goyal, “Performance of a Network,” *GeeksforGeeks*, 27-Sep-2021. [Online]. Available: https://www.geeksforgeeks.org/performance-of-a-network/. (Accessed: Apr. 3, 2022).

(14) F. Shang, W. Su, Q. Wang, H. Gao, and Q. Fu, “A location estimation algorithm based on RSSI vector similarity degree,” International Journal of Distributed Sensor Networks, vol. 10, no. 8, p. 371350, 2014.

(15) A. H. A. Hume, “What is the formula for reliability?,” *Helpful Students Blog*. [Online]. Available: https://www.sweatlodgeradio.com/what-is-the-formula-for-reliability/. (Accessed: Apr. 3, 2022).

(16) “Methodology overview,” *Opensignal*, 11-Apr-2022. [Online]. Available: https://www.opensignal.com/methodology-overview. (Accessed: Apr. 4, 2022).

(17) “About Us,” *About - nPerf.com*. [Online]. Available: https://www.nperf.com/en/about-us/. (Accessed: Apr. 5, 2022).

(18) “Internet speed test,” *Internet Speed test : Test your broadband connection - nPerf.com*. [Online]. Available: https://www.nperf.com/en/. (Accessed: Apr. 6, 2022).

(19) “About Speedtest,” *Speedtest by Ookla - The Global Broadband Speed Test*. [Online]. Available: https://www.speedtest.net/about. (Accessed: Apr. 8, 2022).

APPENDIX

A.1 Data Sample

In block A

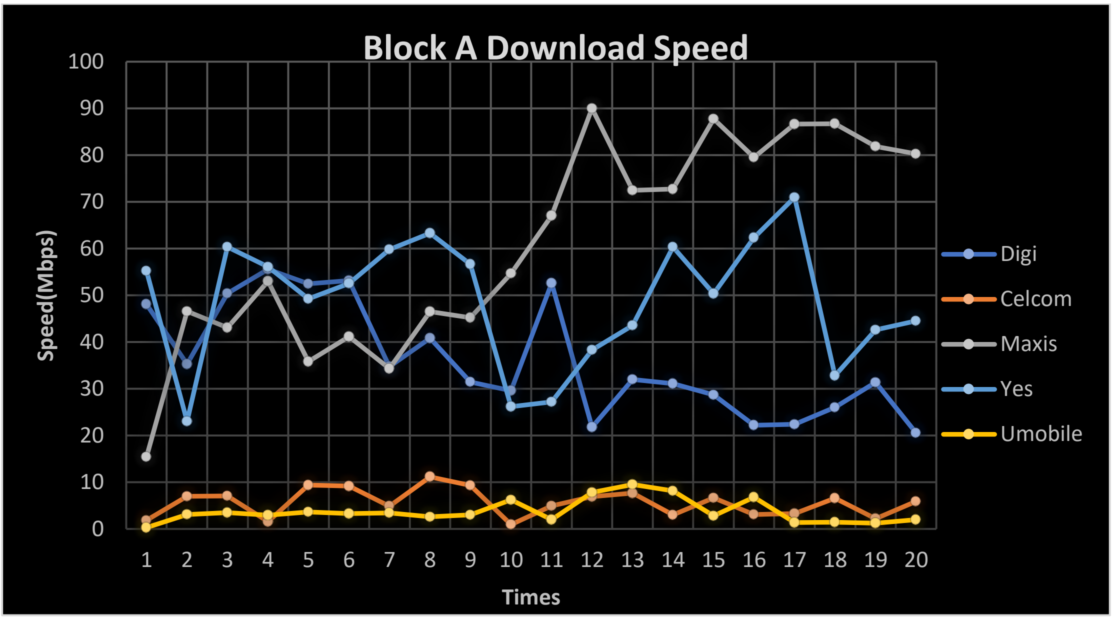


Figure 8.1 Twenty Times Download Speed Test Result in Block A

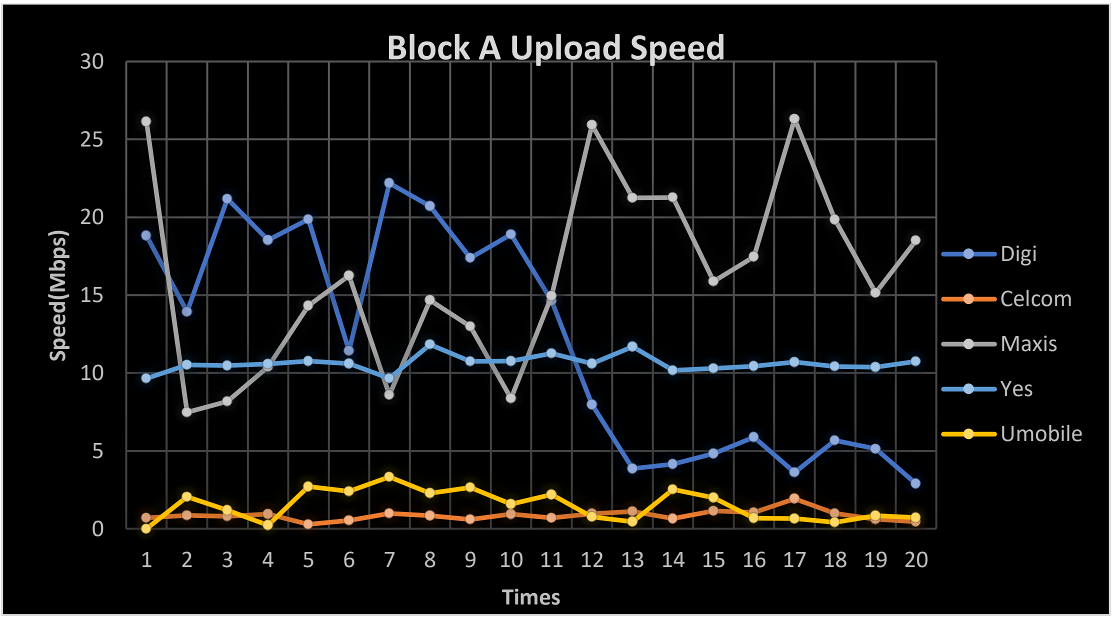


Figure 8.2 Twenty Times Upload Speed Test Result in Block A

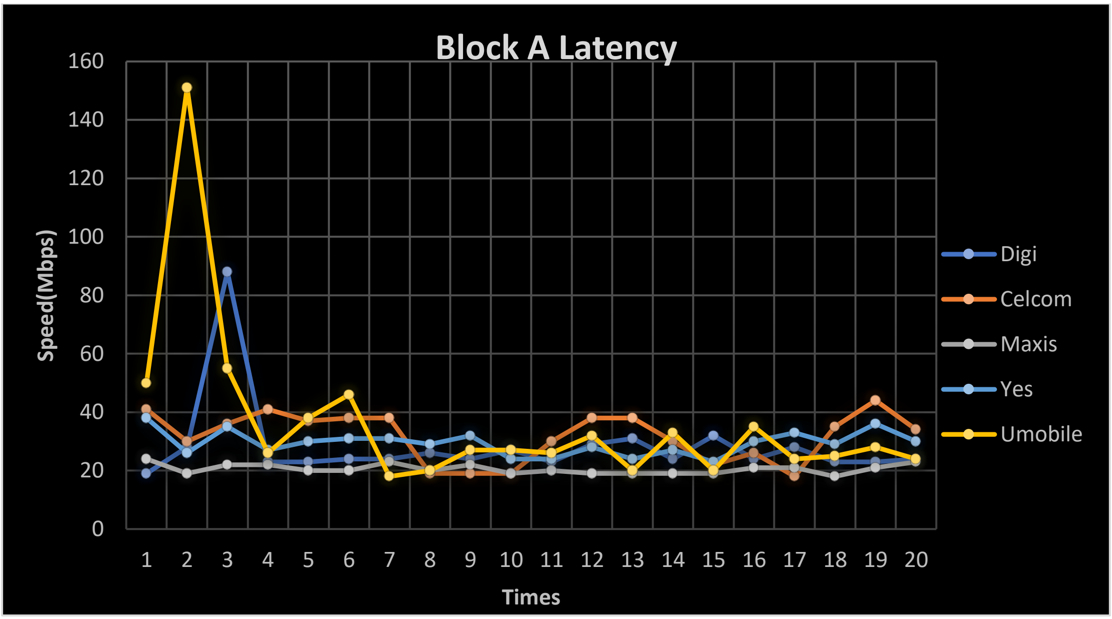


Figure 8.3 Twenty Times Latency Test Result in Block A

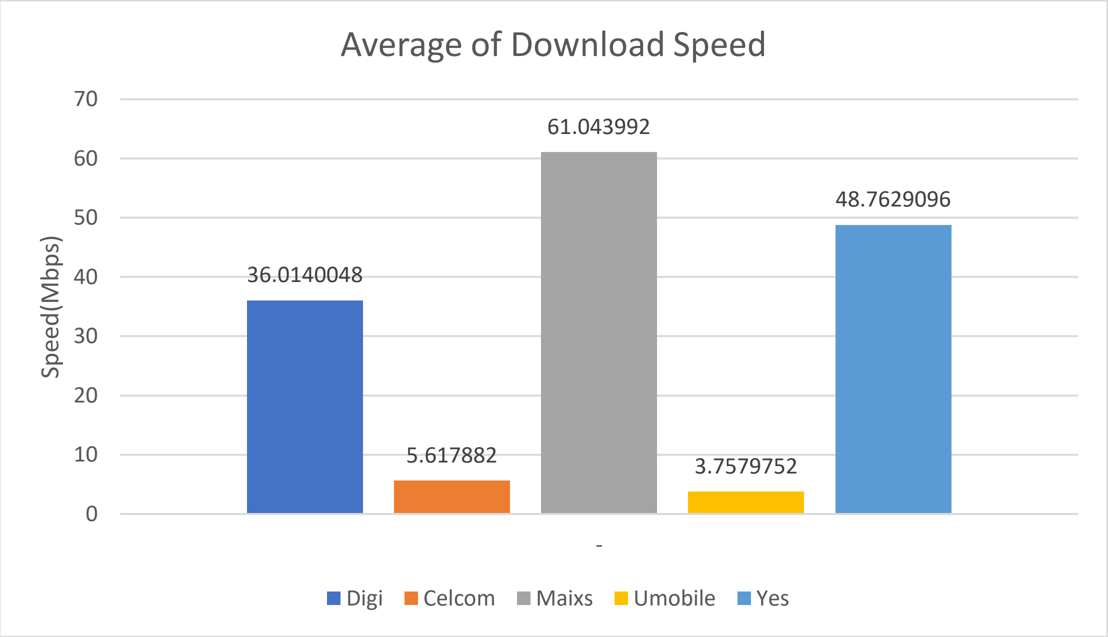


Figure 8.4 Average Download Speed in Block A

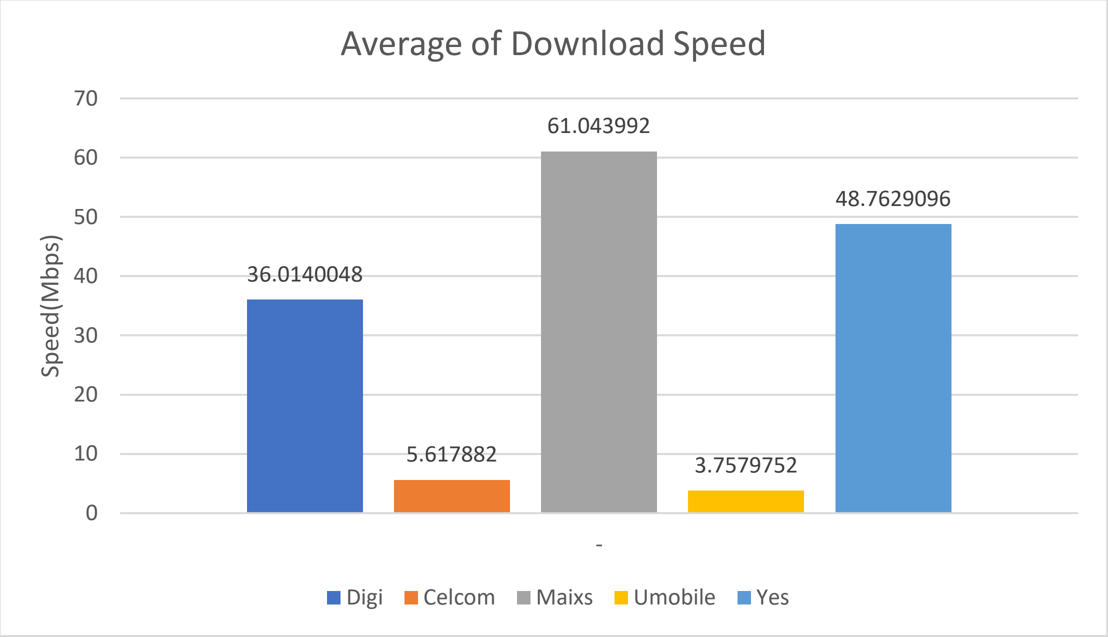


Figure 8.5 Average Upload Speed in Block A

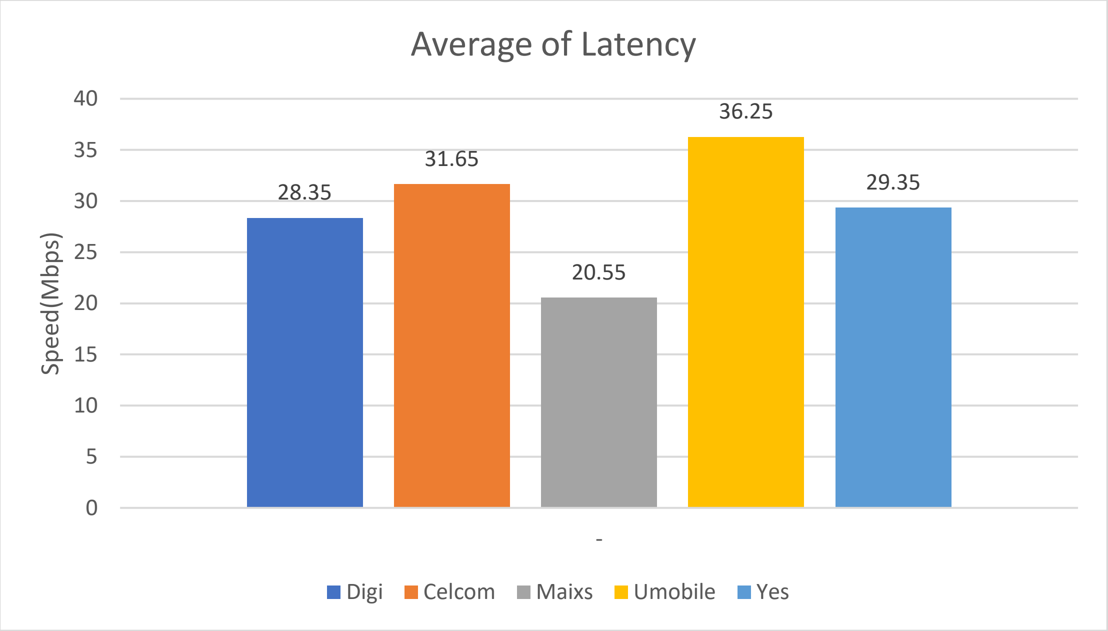


Figure 8.6 Average Latency in Block A

In block B&C

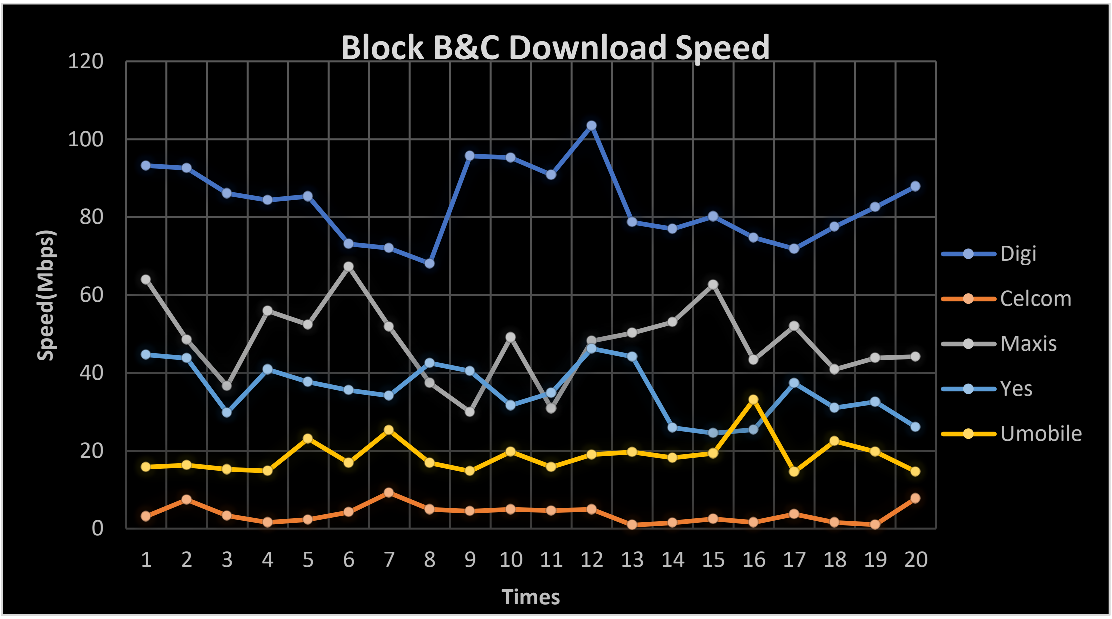


Figure 8.7 Twenty Times Download Speed Test Result in Block A

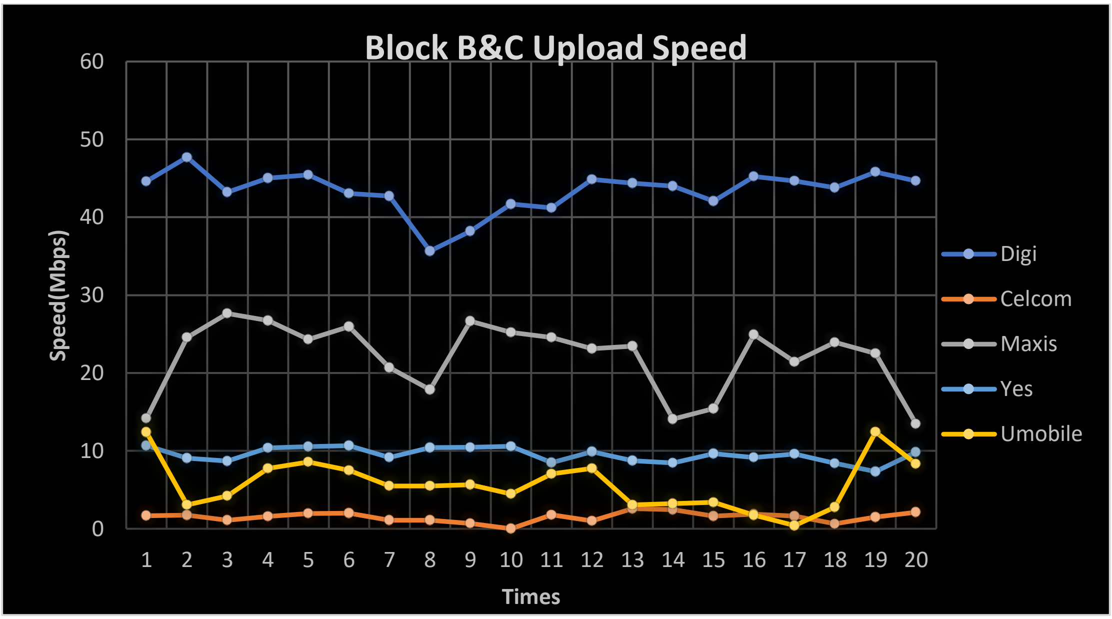


Figure 8.8 Twenty Times Upload Speed Test Result in Block A

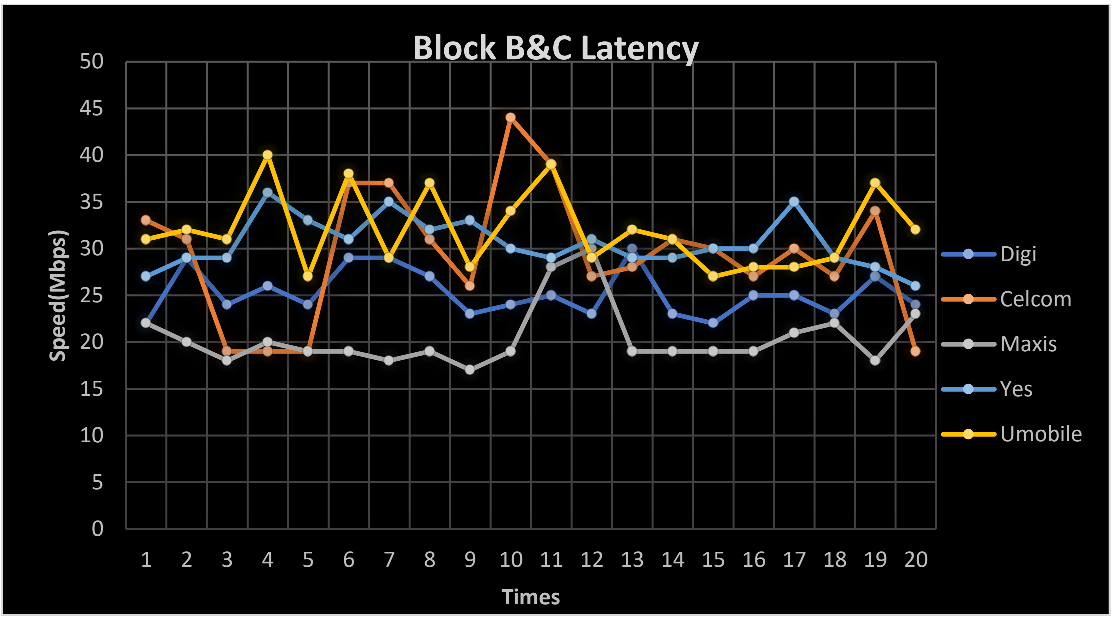


Figure 8.9 Twenty Times Latency Test Result in Block A

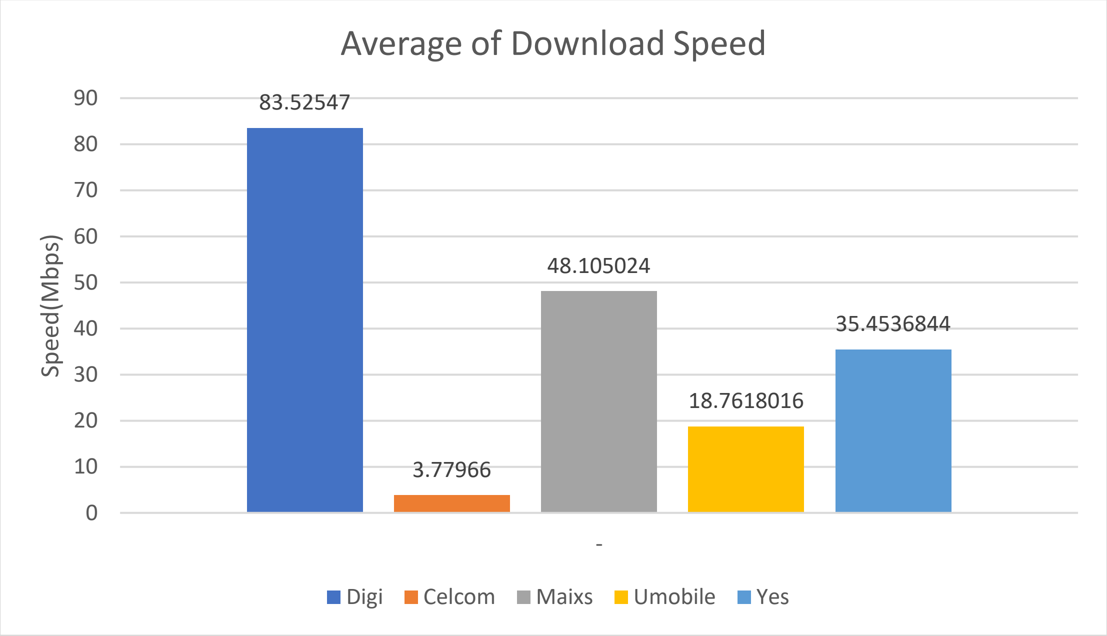


Figure 8.10 Average Download Speed in Block B&C

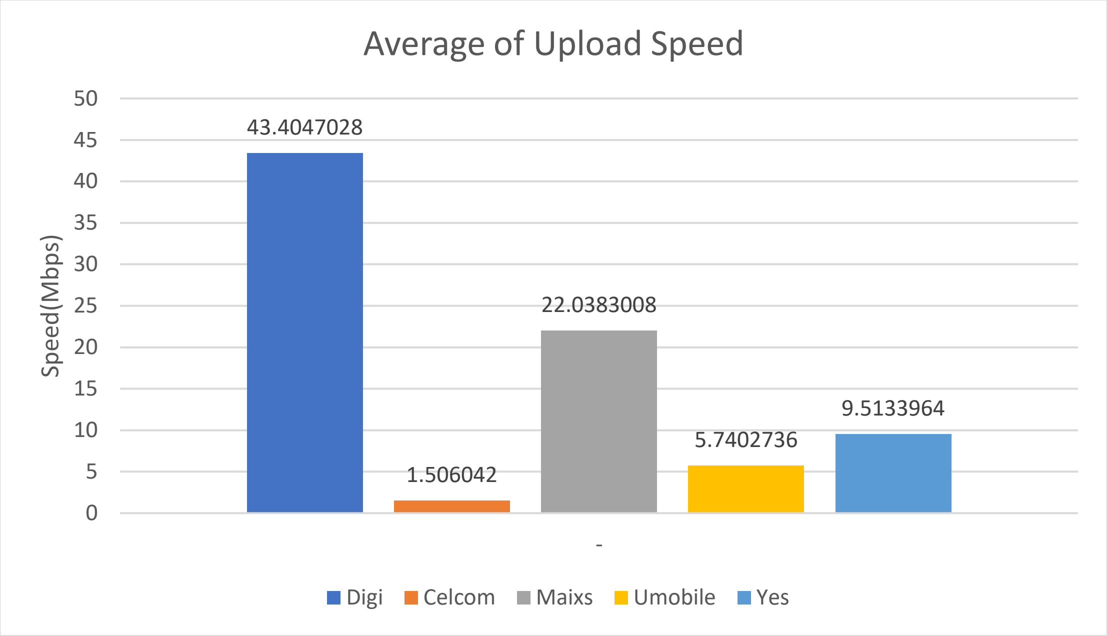


Figure 8.11 Average Uplaod Speed in Block B&C

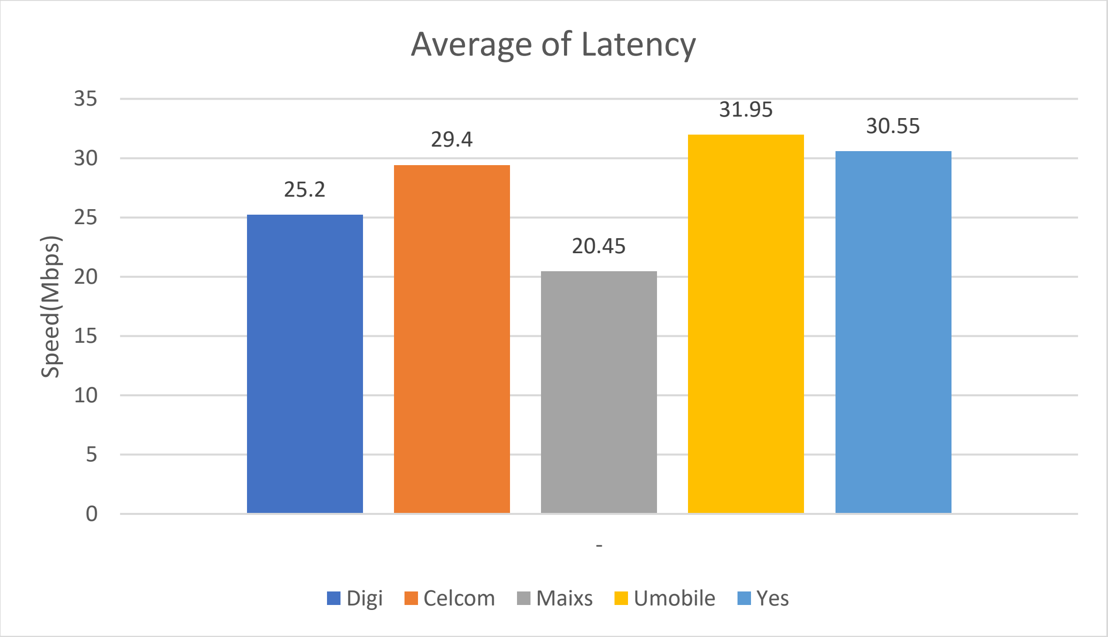


Figure 8.12 Average Latency in Block B&C

In Block D&E

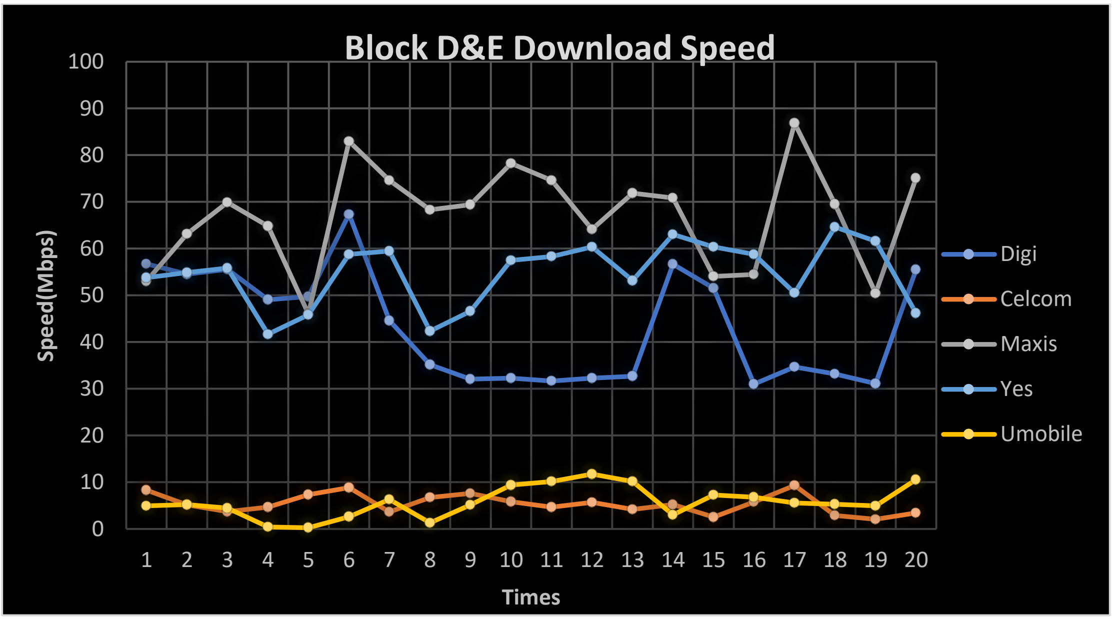


Figure 8.13 Twenty Times Download Speed Test Result in Block D&E

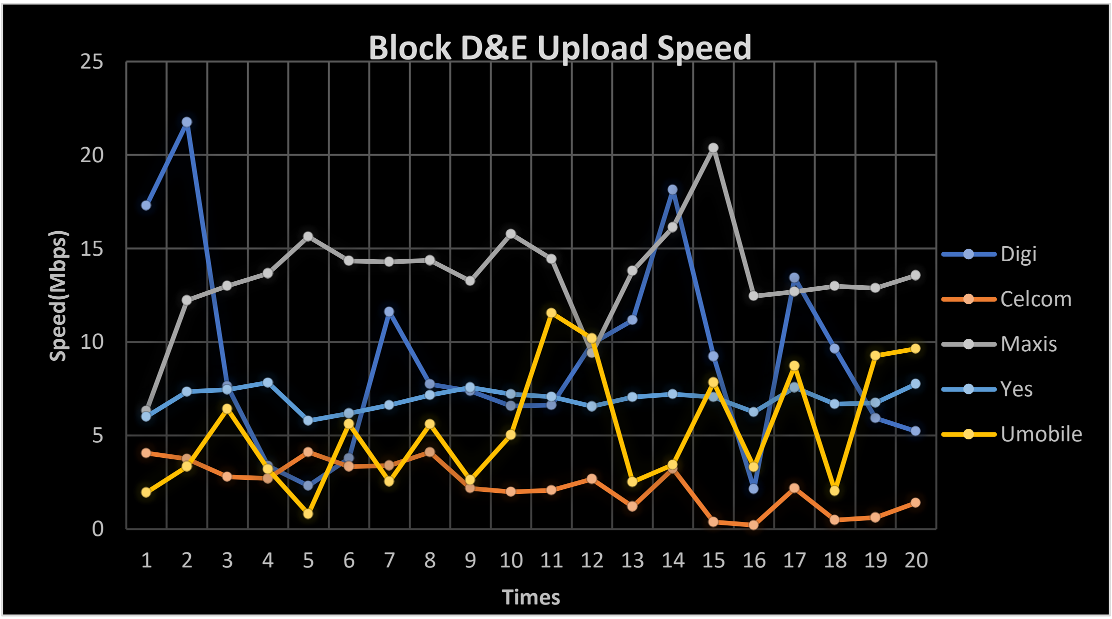


Figure 8.14 Twenty Times Upload Speed Test Result in Block D&E

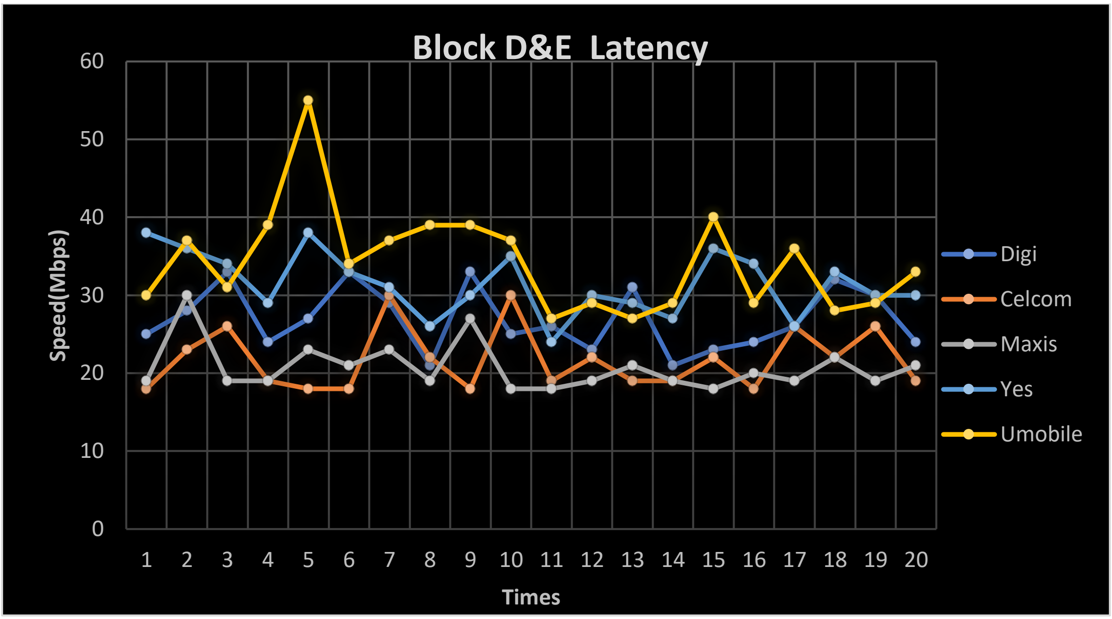


Figure 8.15 Twenty Times Latency Test Result in Block D&E

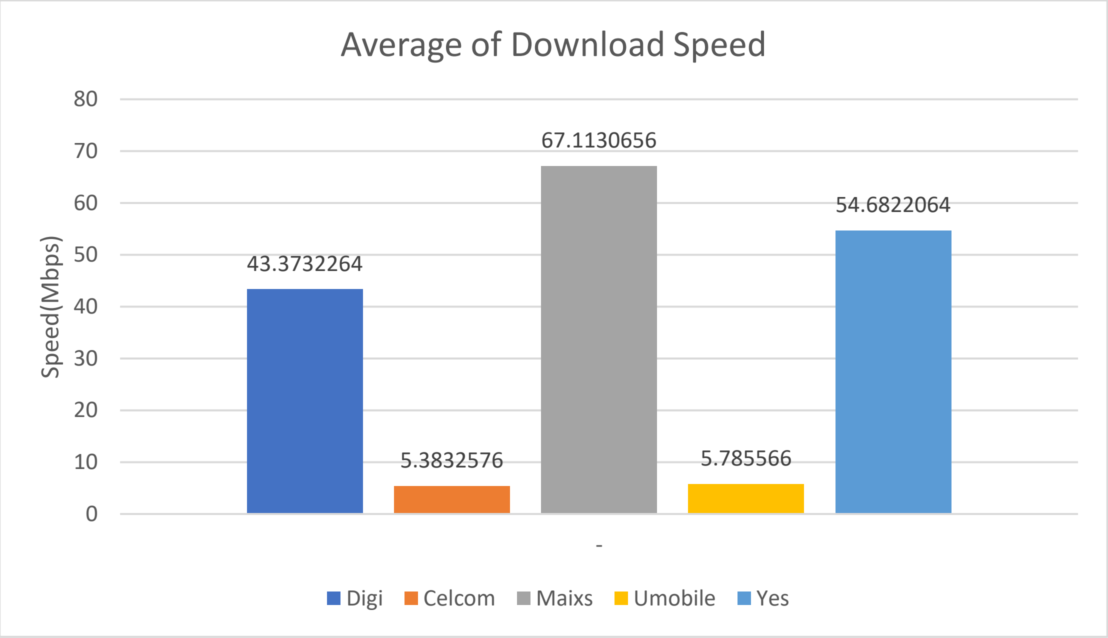


Figure 8.16 Average Download Speed in Block D&E

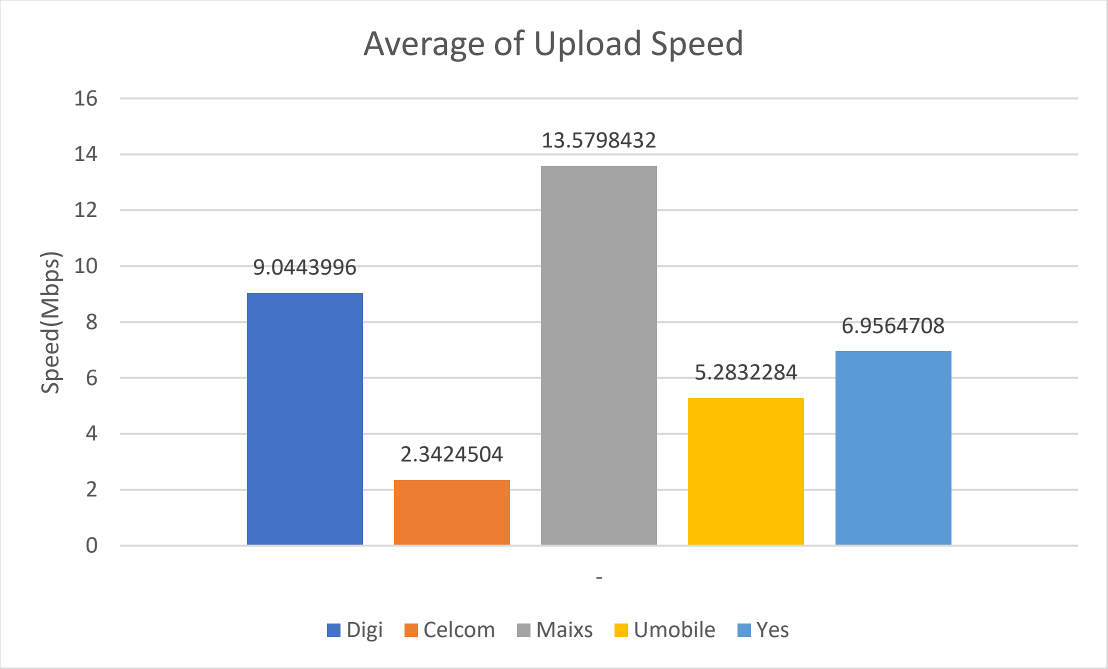


Figure 8.17 Average Upload Speed in Block D&E

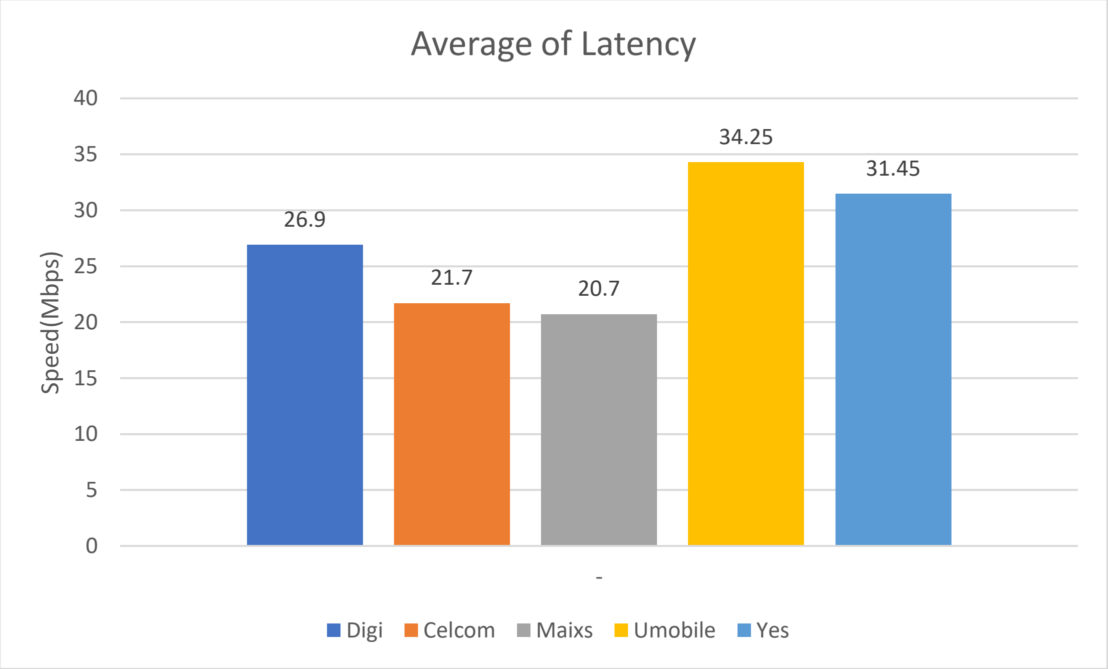


Figure 8.18 Average Latency in Block D&E

In Block F&G&H&I

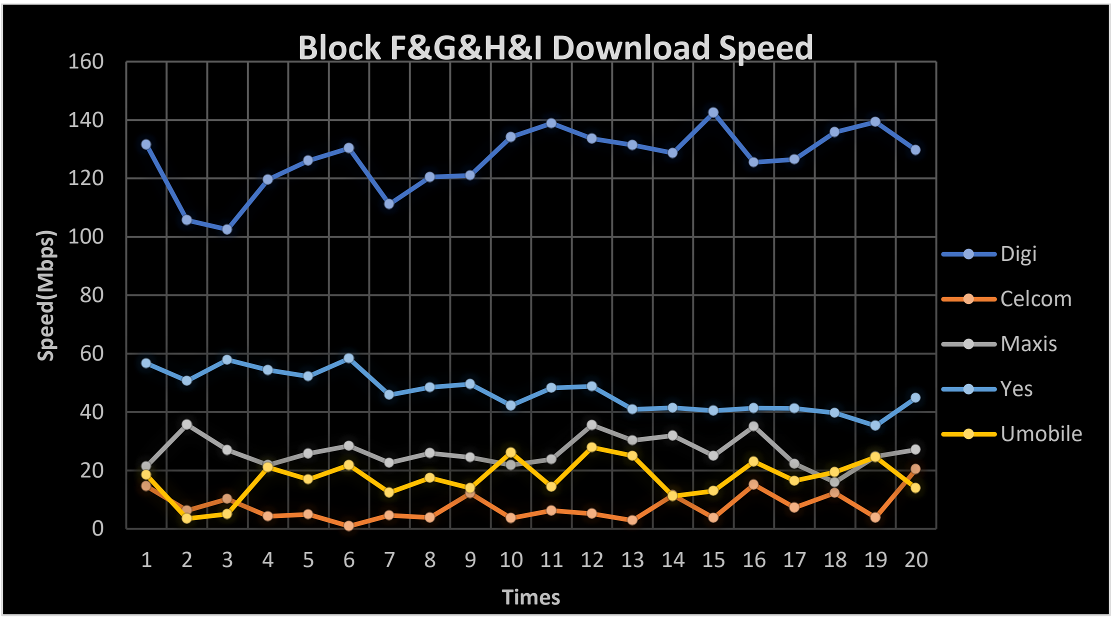


Figure 8.19 Twenty Times Download Speed Test Result in Block F&G&H&I

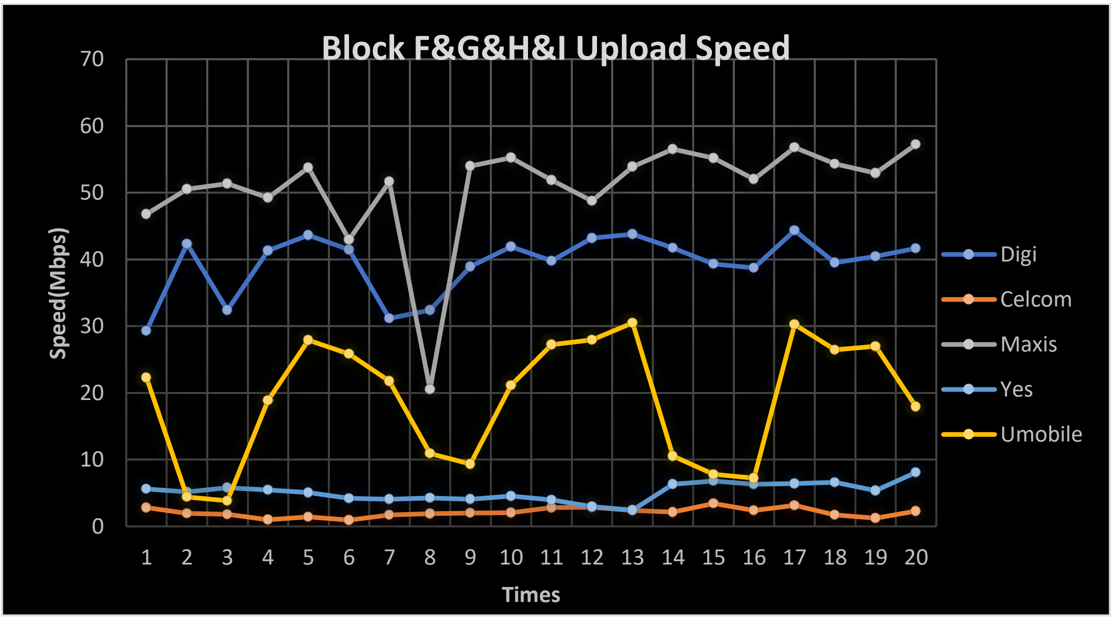


Figure 8.20 Twenty Times Upload Speed Test Result in Block F&G&H&I

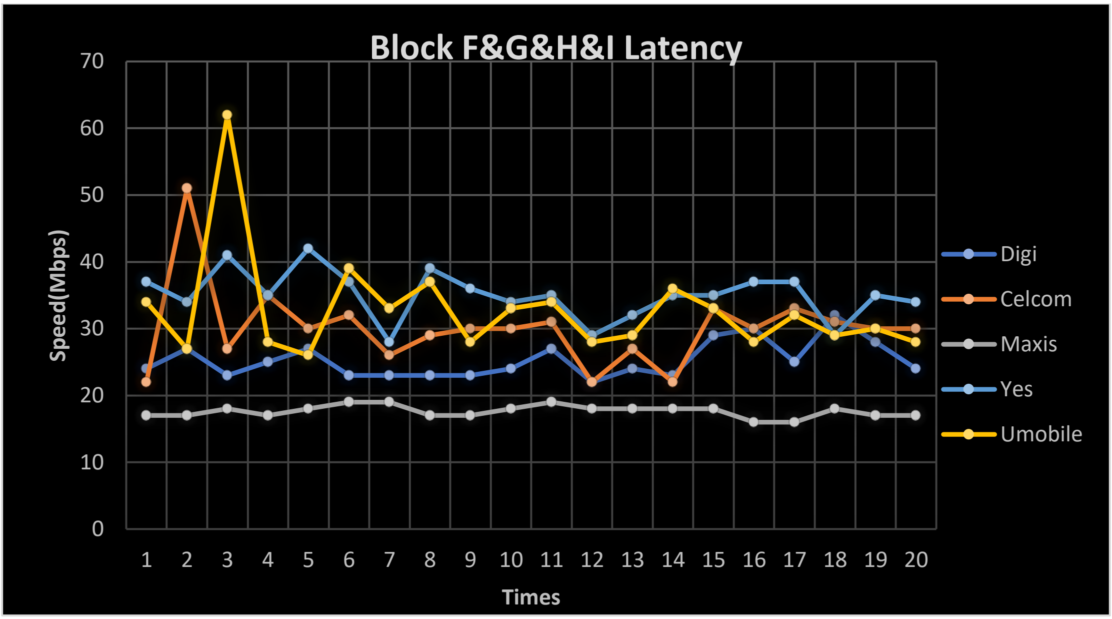


Figure 8.21 Twenty Times Latency Test Result in Block F&G&H&I

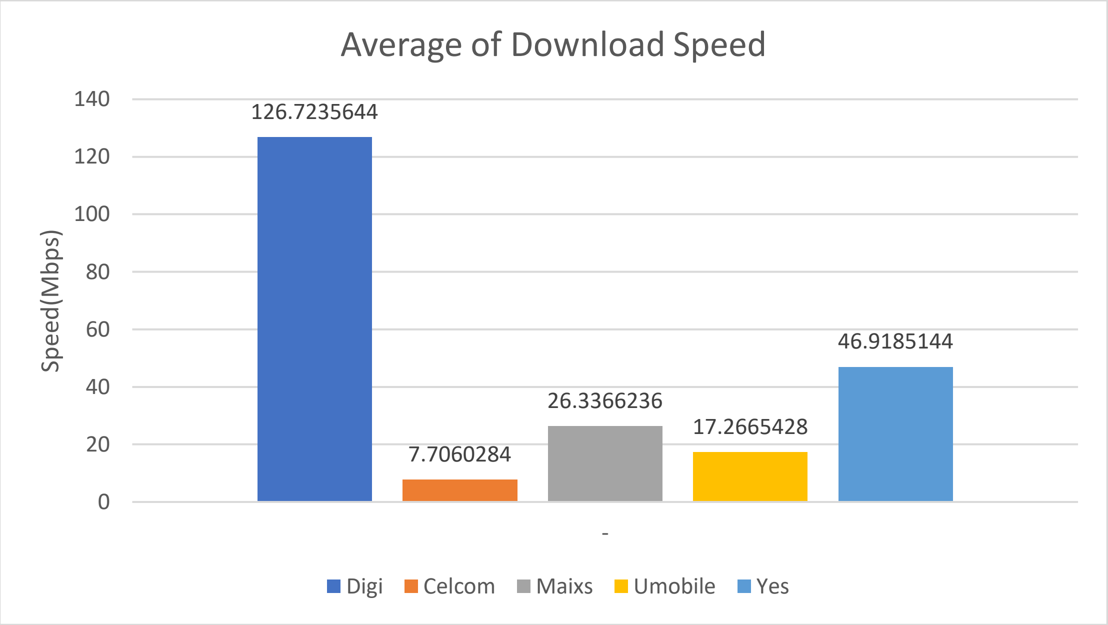


Figure 8.22 Average Download Speed in Block F&G&H&I

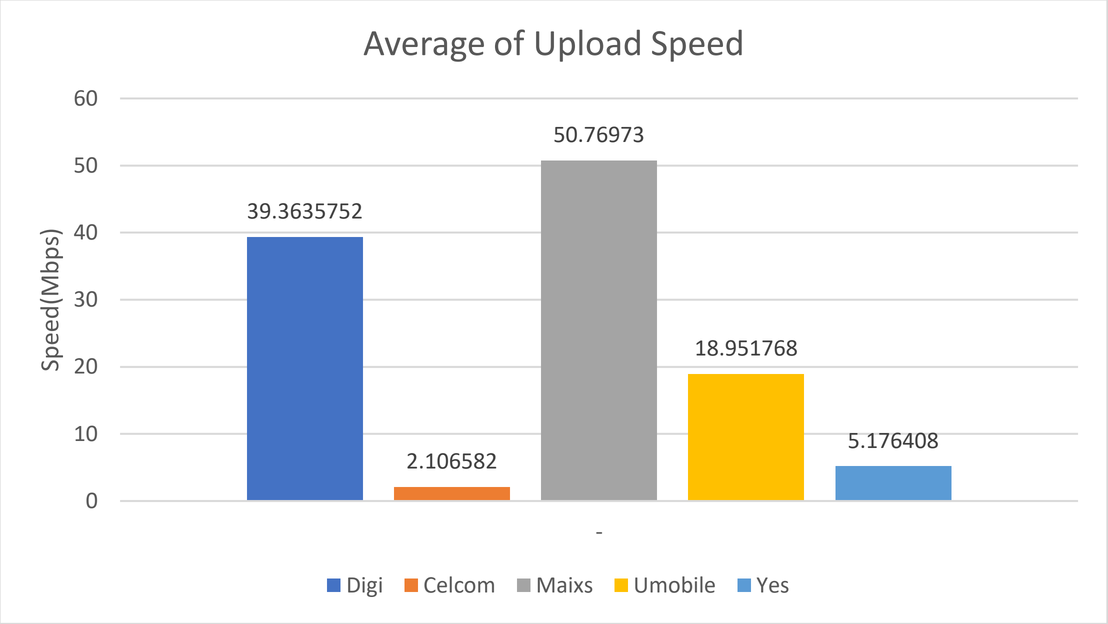


Figure 8.23 Average Upload Speed in Block F&G&H&I

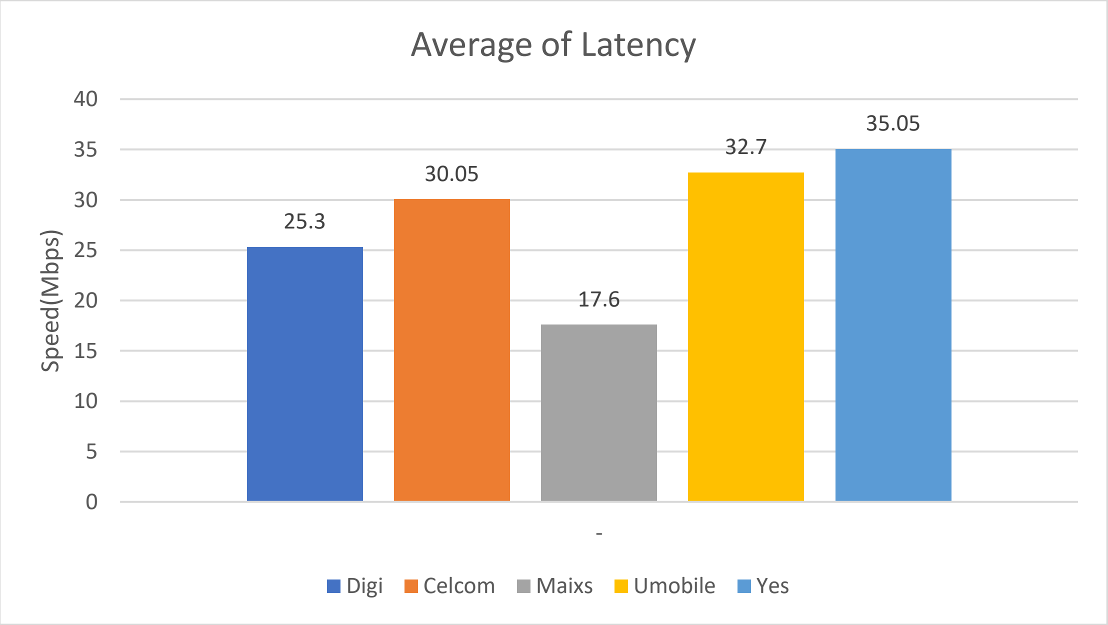


Figure 8.24 Average Latency in Block F&G&H&I

In Block J&K&L

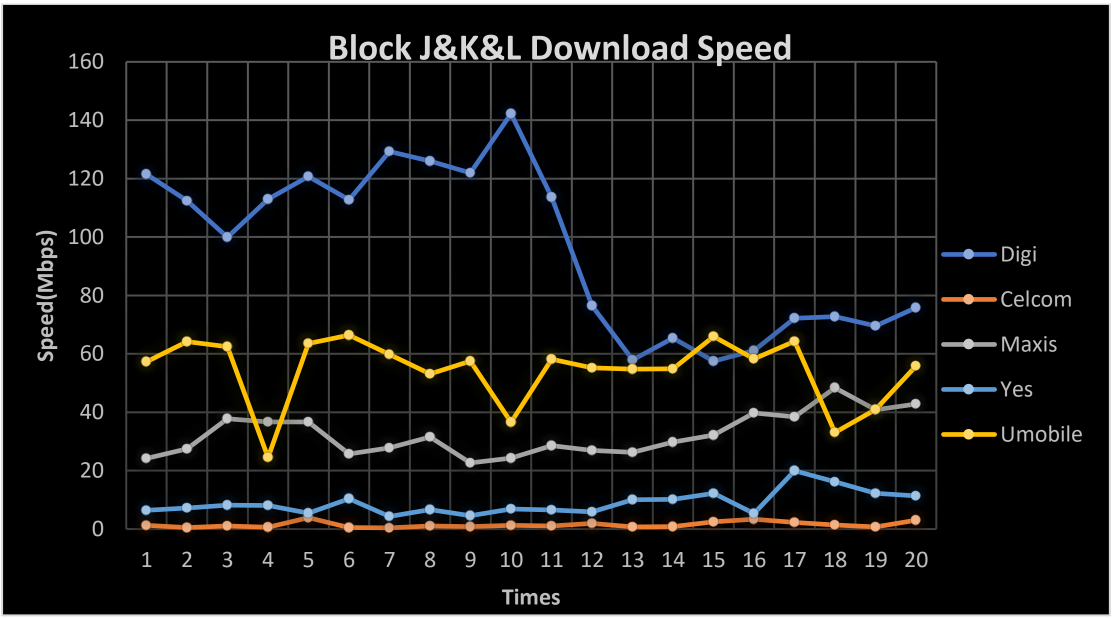


Figure 8.25 Twenty Times Download Speed Test Result in Block J&K&L

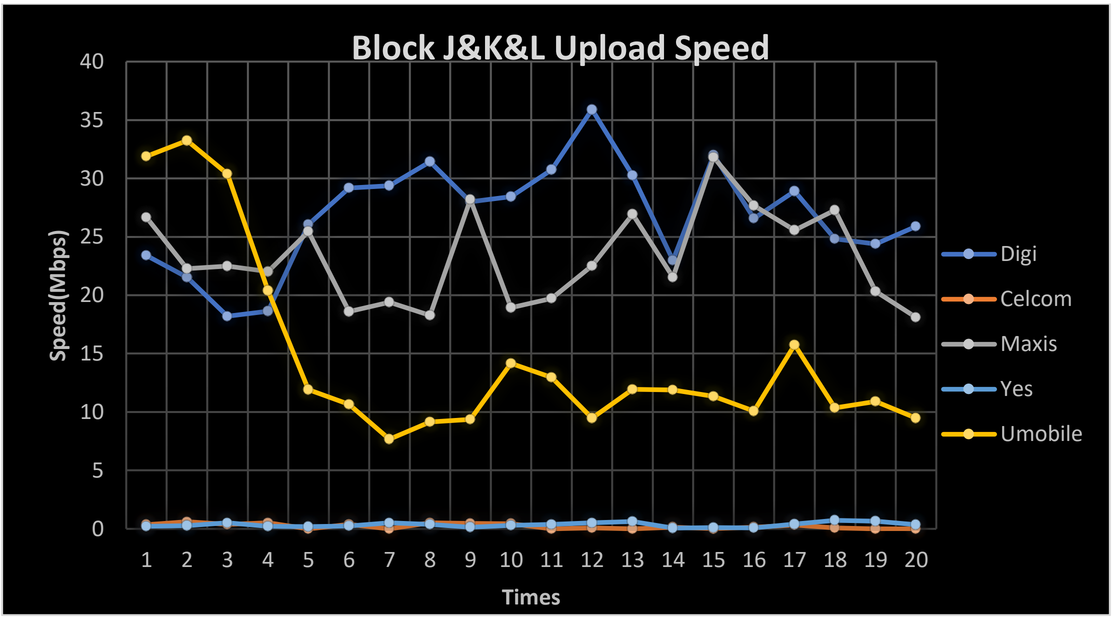


Figure 8.26 Twenty Times Upload Speed Test Result in Block J&K&L

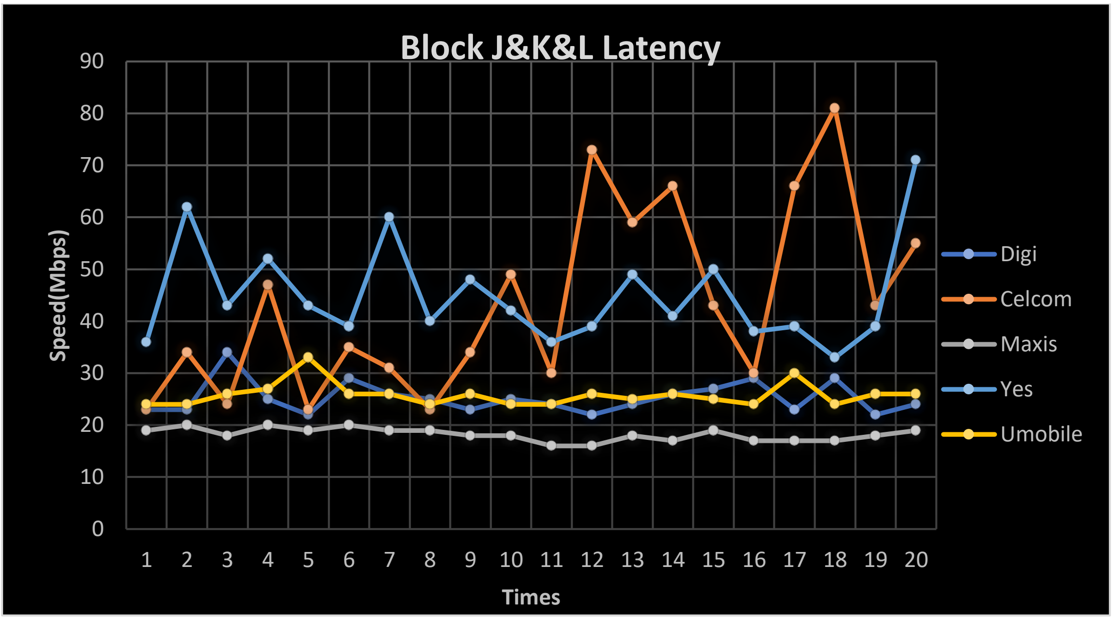


Figure 8.27 Twenty Times Latency Test Result in Block J&K&L

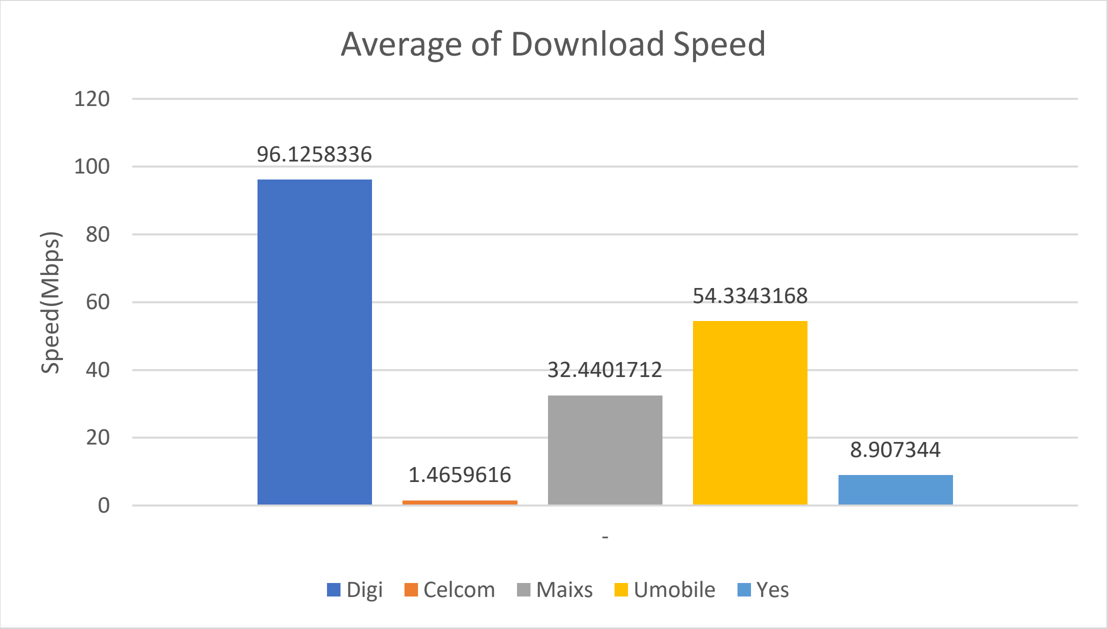


Figure 6.28 Average Download Speed in Block J&K&L

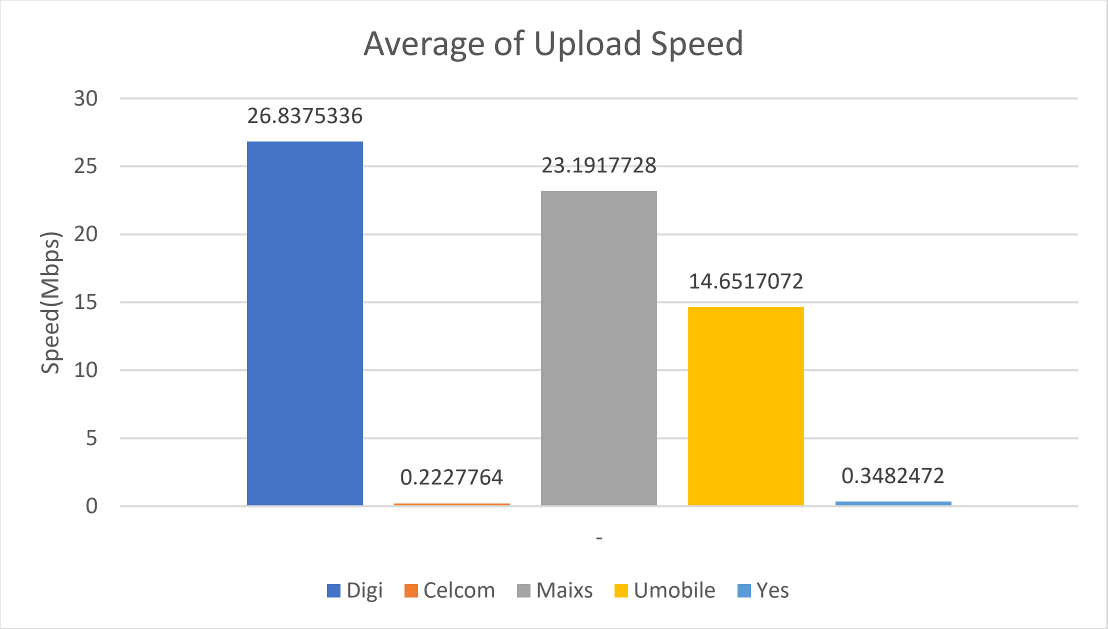


Figure 8.29 Average Upload Speed in Block J&K&L

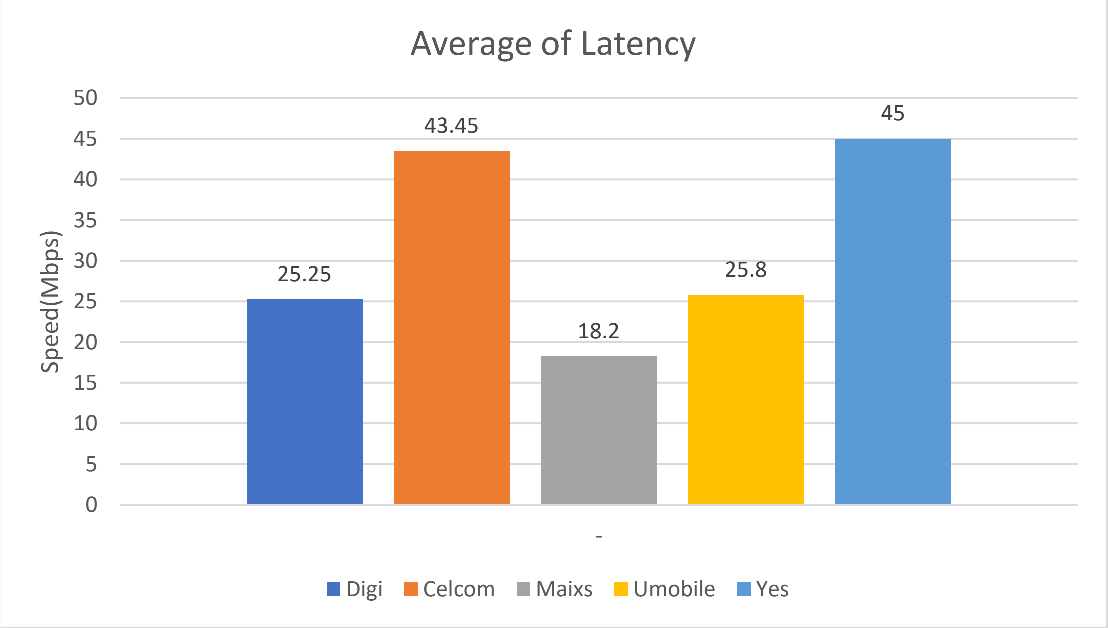


Figure 8.30 Average Latency in Block J&K&L

In Block M

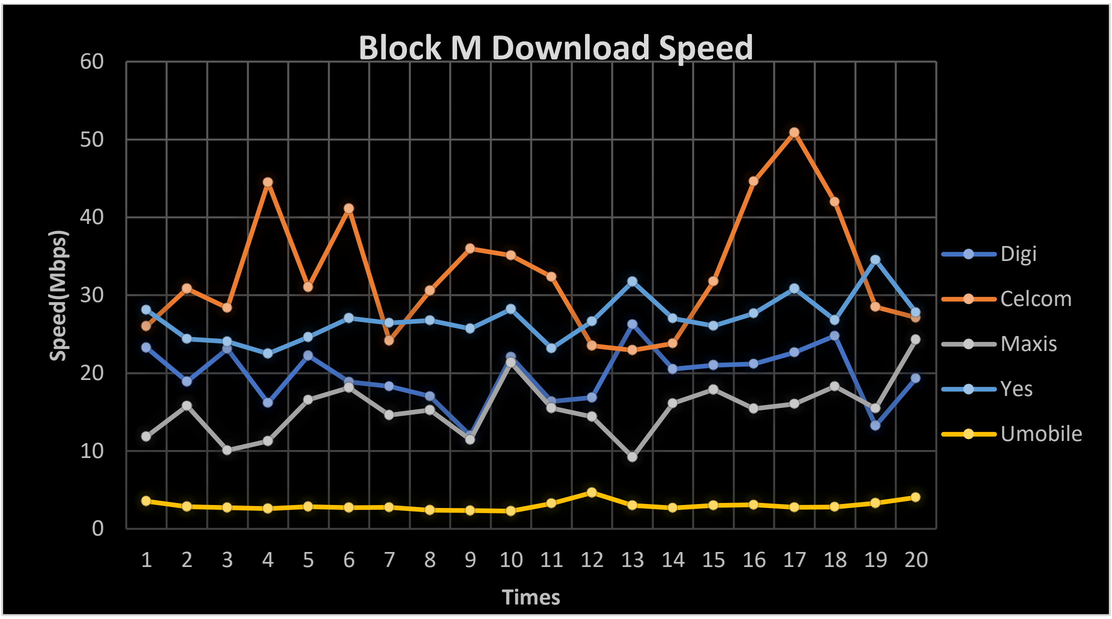


Figure 8.31 Twenty Times Download Speed Test Result in Block M

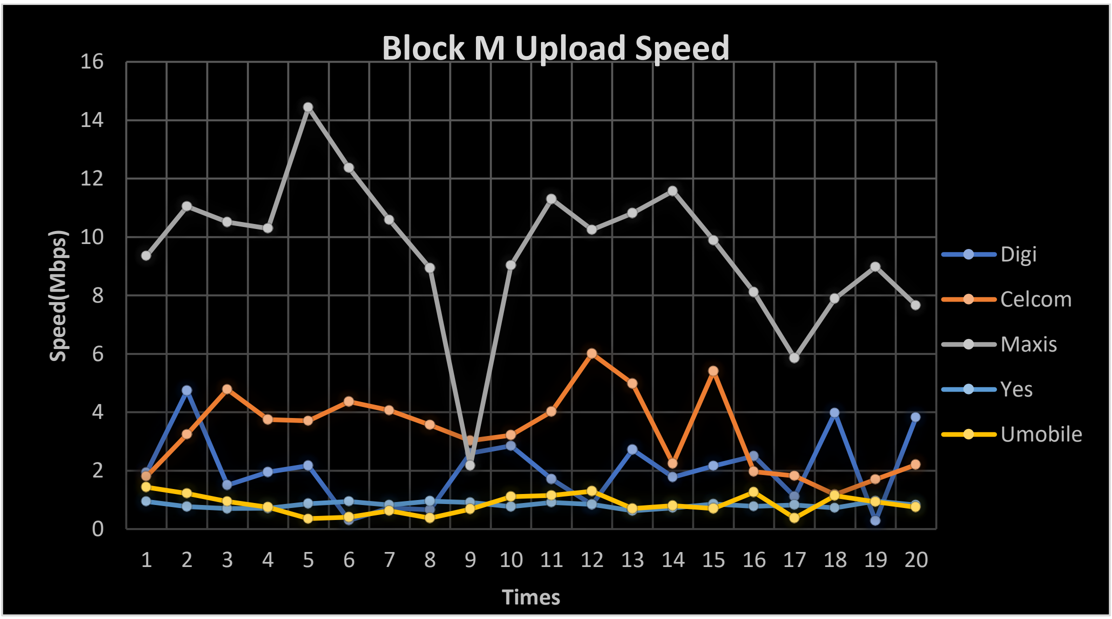


Figure 8.32 Twenty Times Upload Speed Test Result in Block M

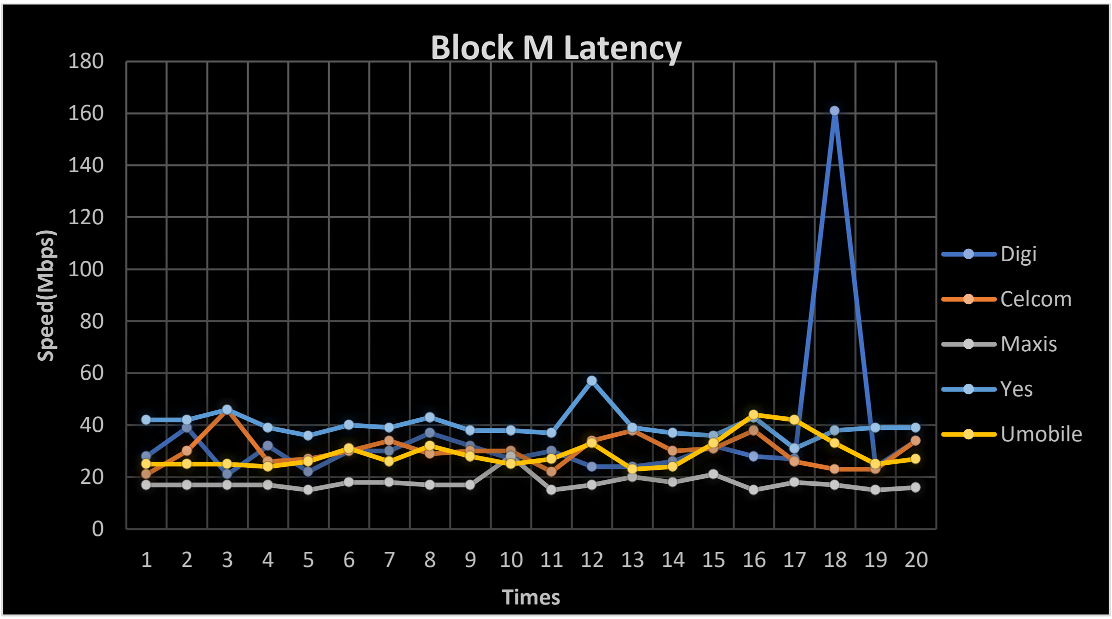


Figure 8.33 Twenty Times Latency Test Result in Block M

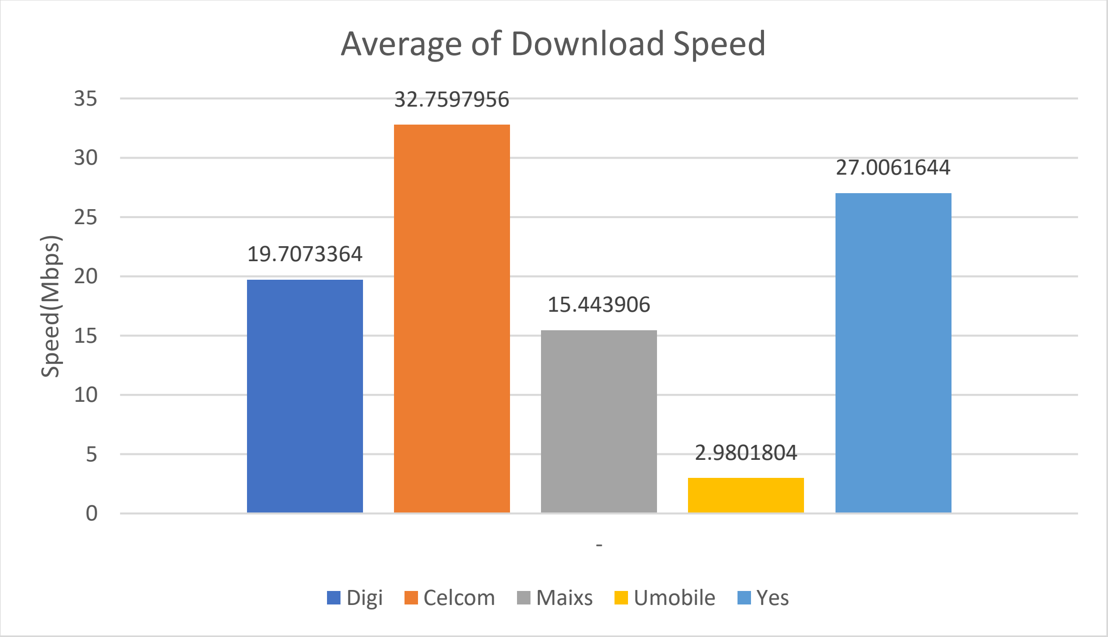


Figure 8.34 Average Download Speed in Block M

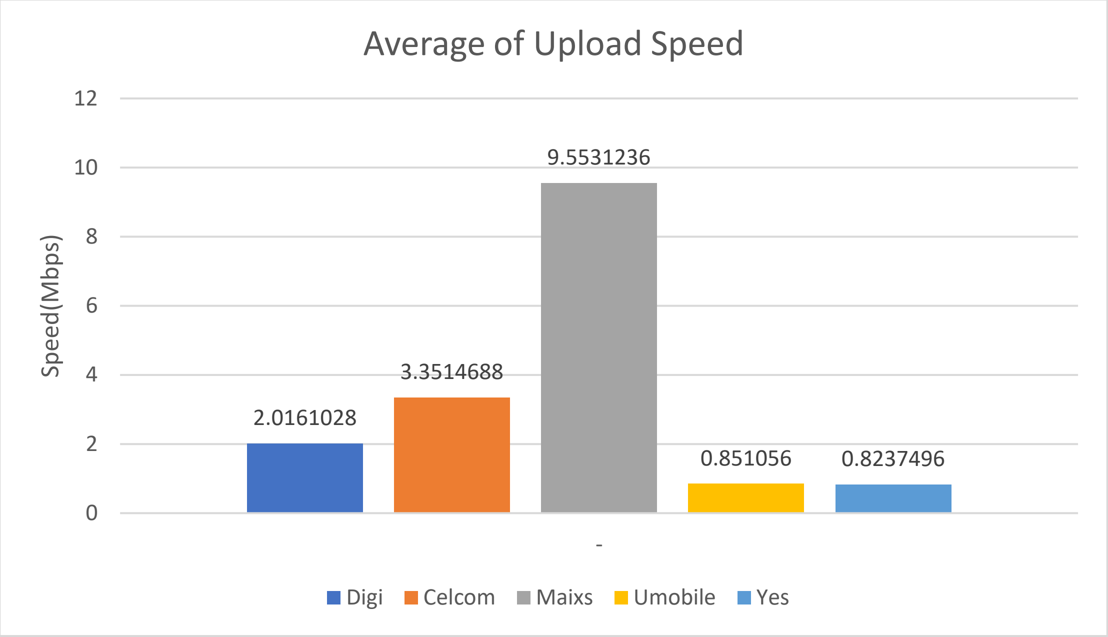


Figure 8.35 Average Upload Speed in Block M

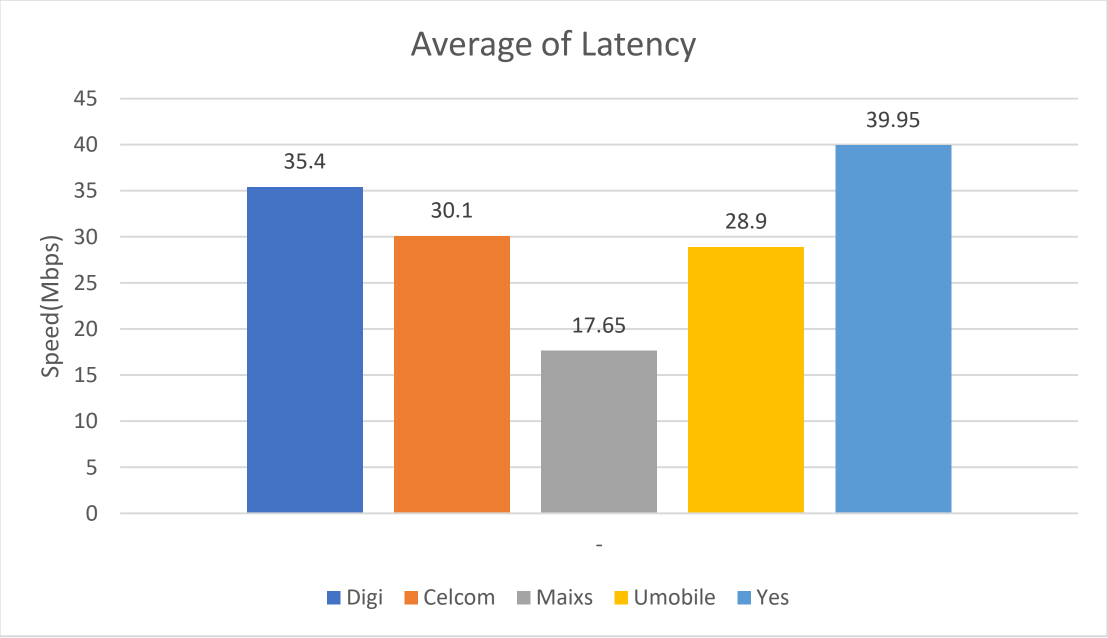


Figure 8.36 Latency Speed in Block M

In Block N&P

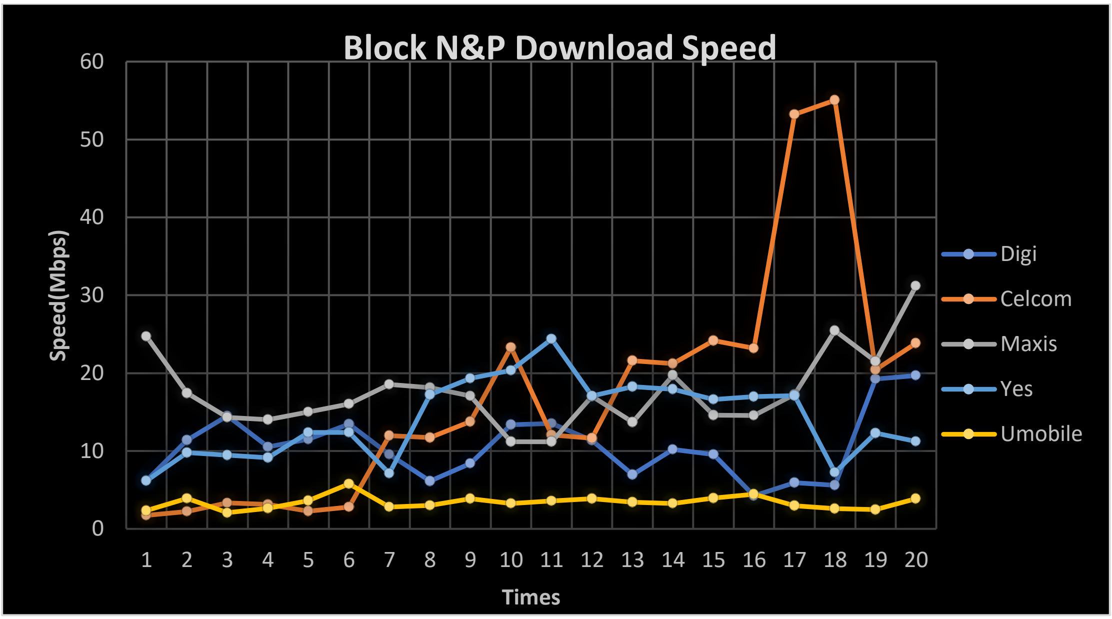


Figure 8.37 Twenty Times Download Speed Test Result in Block N&P

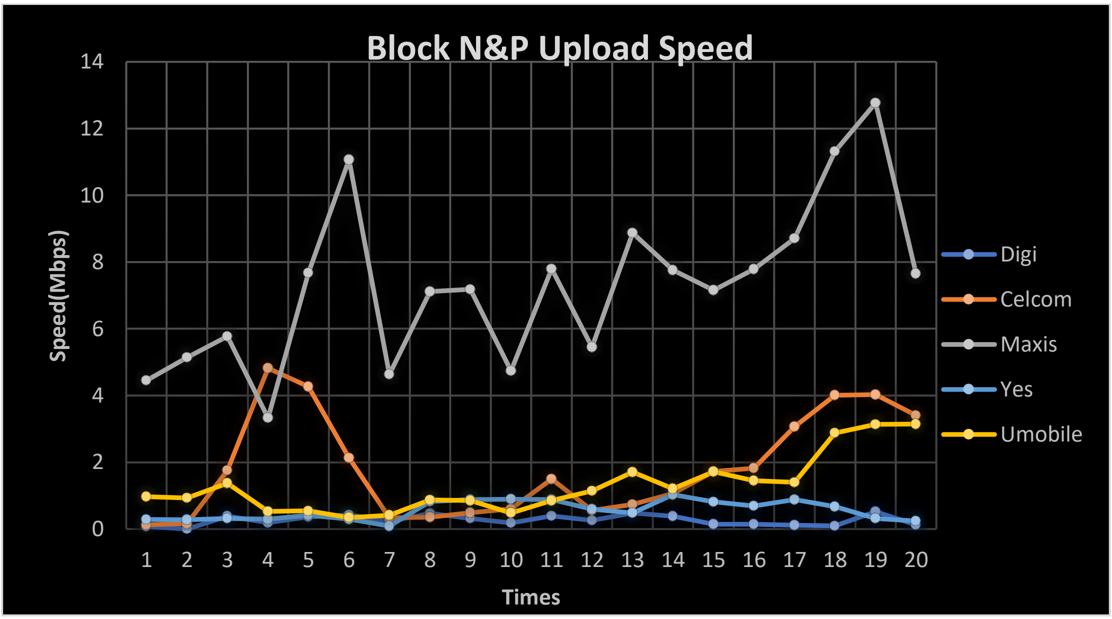


Figure 8.38 Twenty Times Upload Speed Test Result in Block N&P

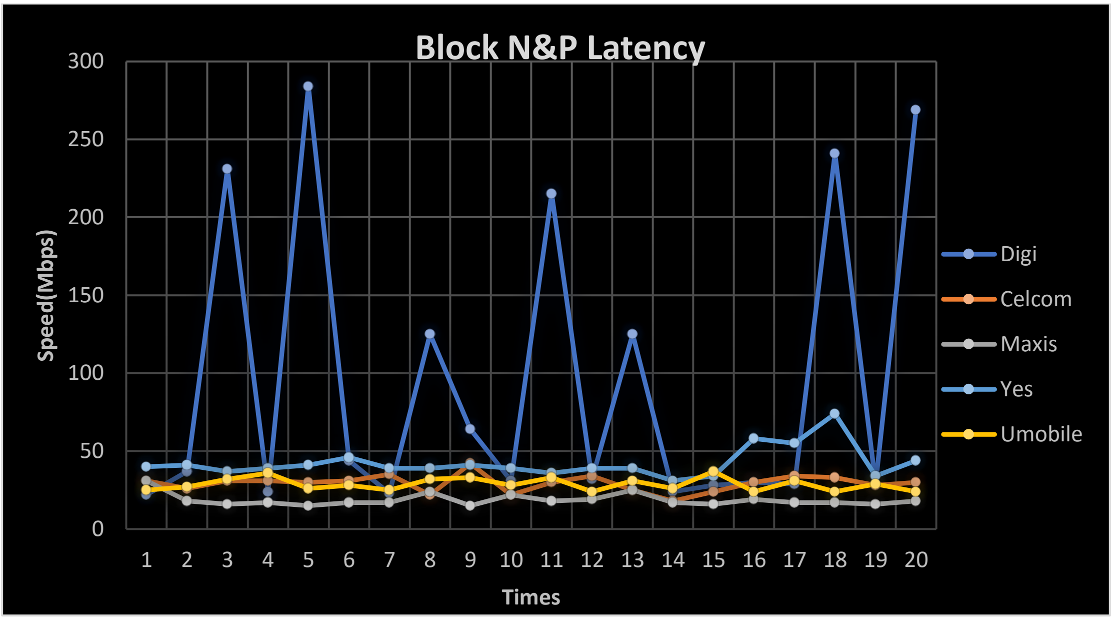


Figure 8.39 Twenty Times Latency Test Result in Block N&P

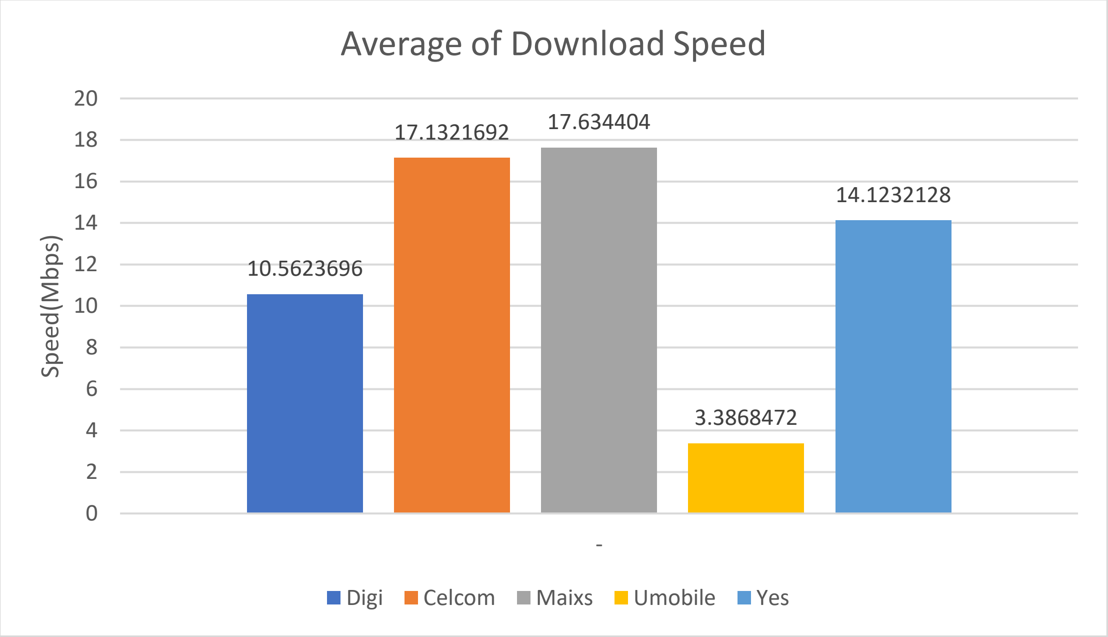


Figure 8.40 Average Download Speed in Block N&P

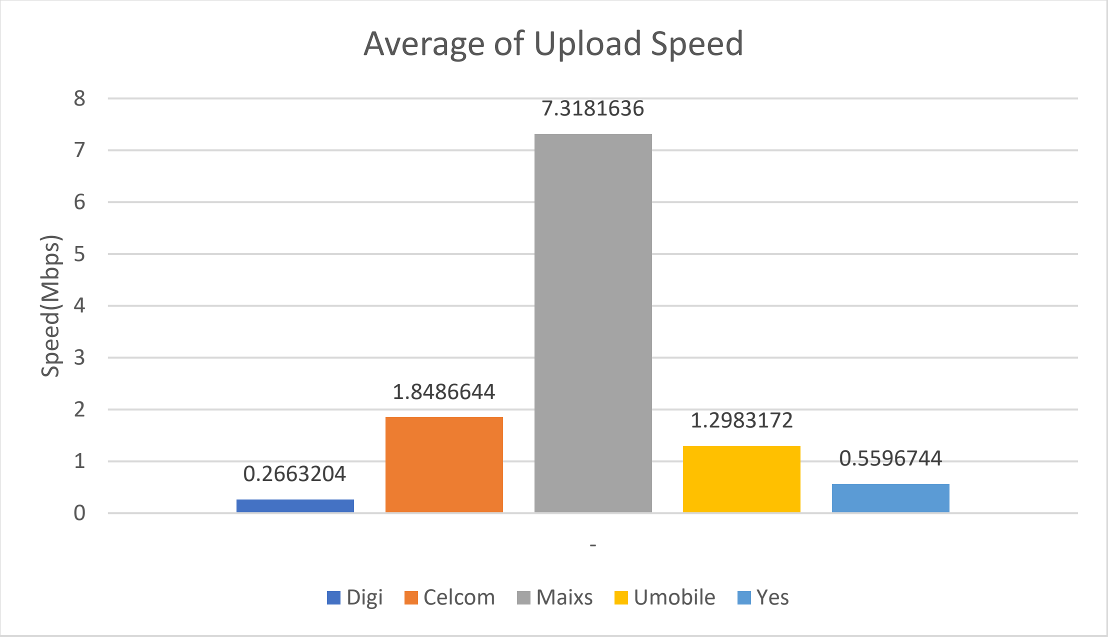


Figure 8.41 Average Upload Speed in Block N&P

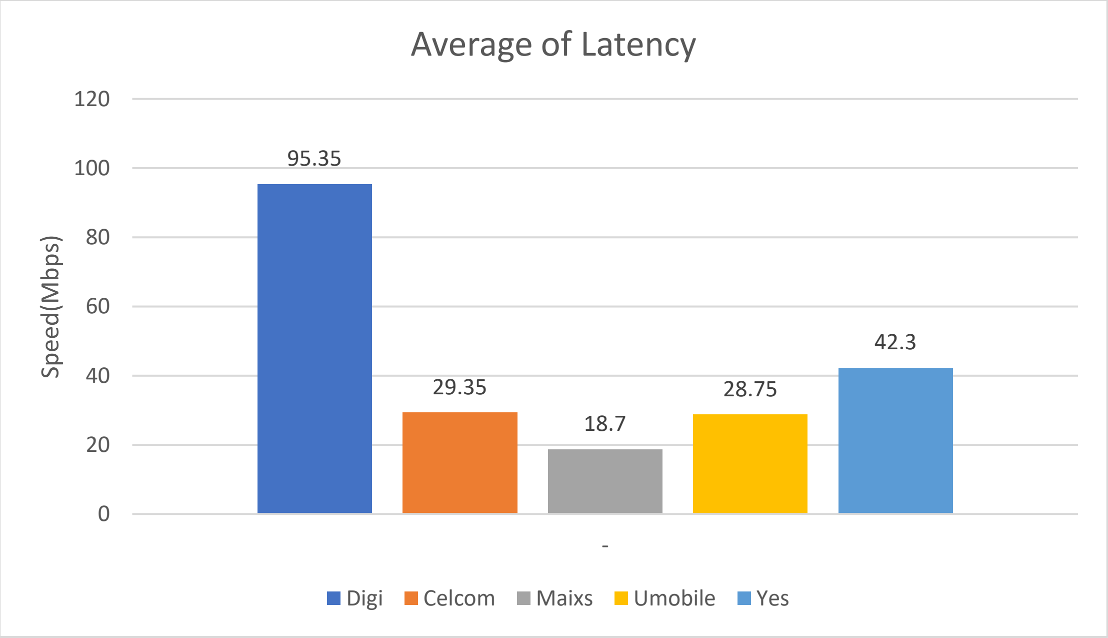


Figure 8.42 Average Latency in Block N&P

Average Download Speed in Kampar Hot Spot Area

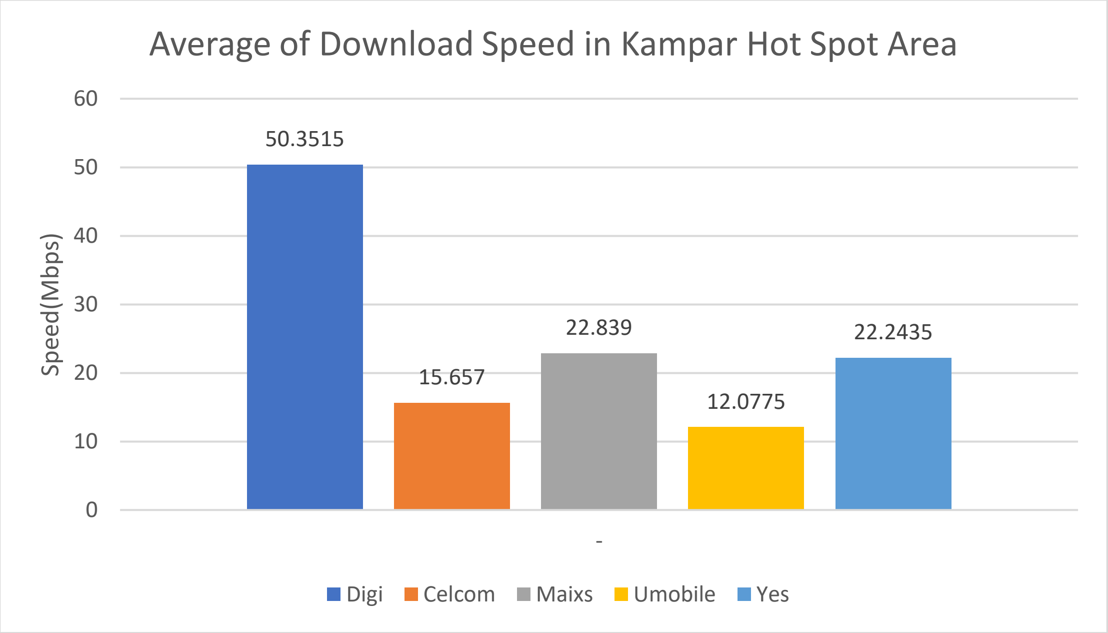


Figure 8.43 Average Download Speed in Kampar Hot Spot Area

Average Upload Speed in Kampar Hot Spot Area

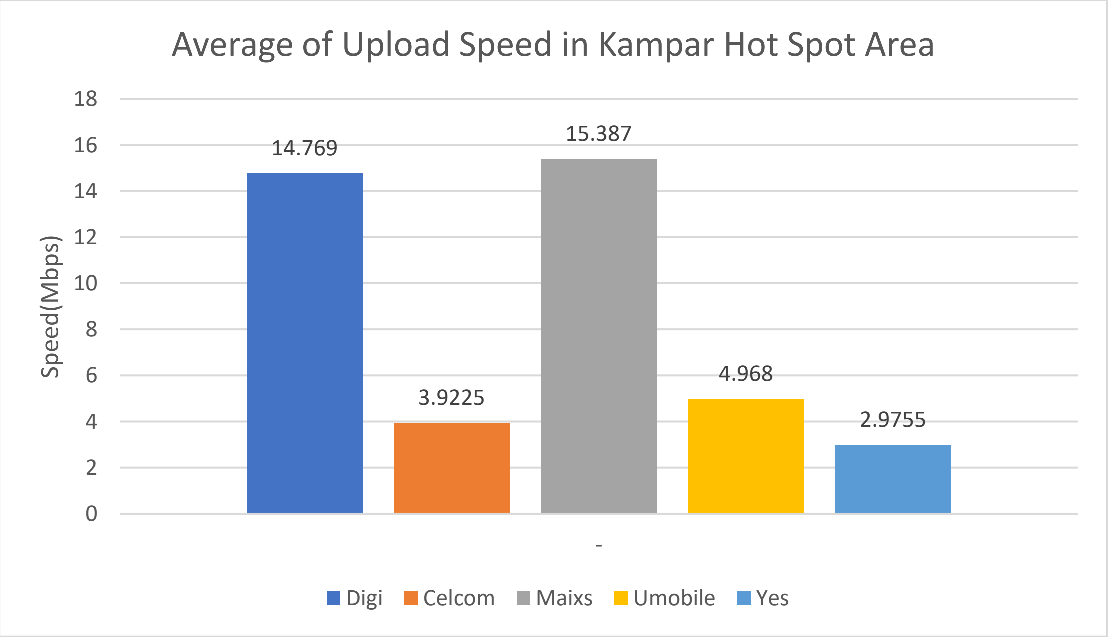


Figure 8.44 Average Upload Speed in Kampar Hot Spot Area

Average Upload Speed in Kampar Hot Spot Area

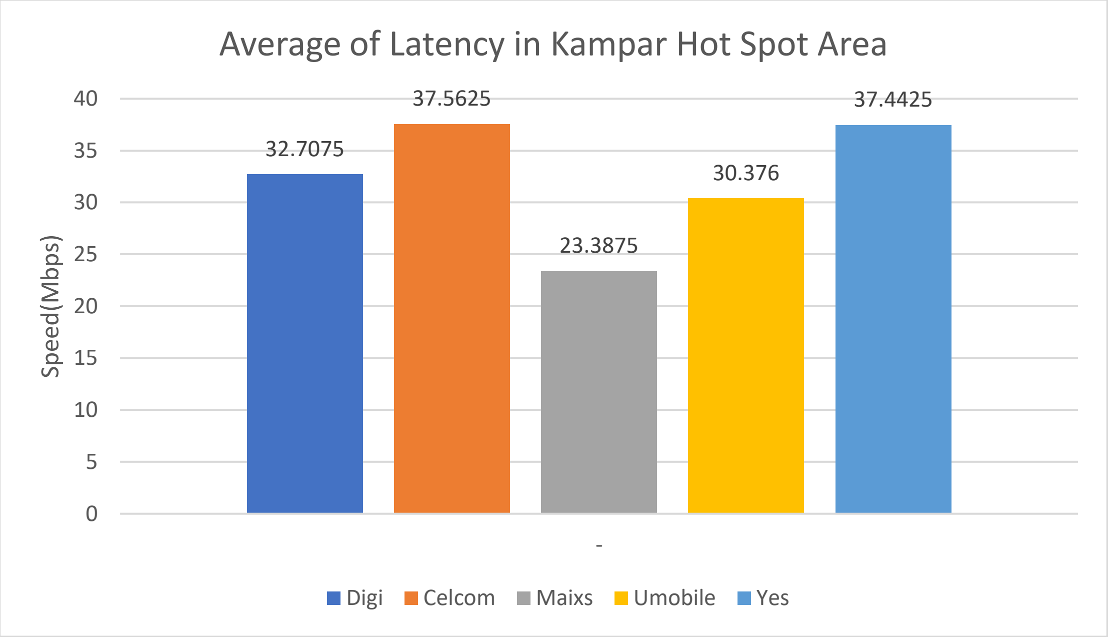


Figure 8.45 Average Latency in Kampar Hot Spot Area

A.2 Inside/Outside UTAR

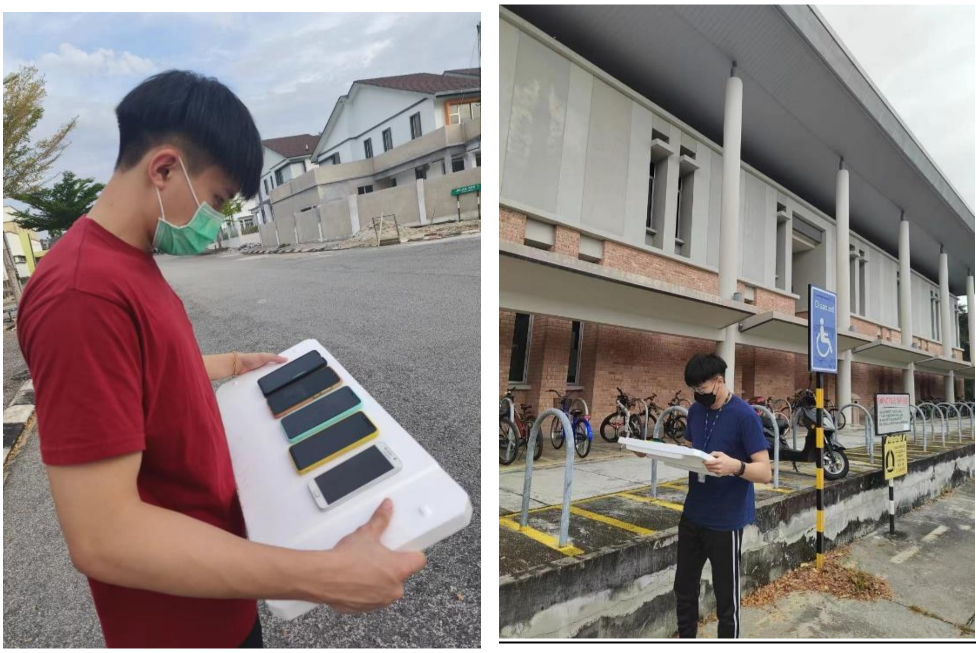


Figure 8.46 Inside and outside of UTAR when collecting data

A.3 Biweekly Report

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 2**
Student Name & ID: Chia Zhen Yik & 19ACB04782	
Supervisor: Dr. Aun Yichiet	
Project Title: **TELCO RECOMMENDATION BASED ON SPEED TEST DATA**	

1. WORK DONE

- Review the FYP II Guideline
- Review back the Collected data in FYP I
- Plan for FYP II

2. WORK TO BE DONE

- Discuss with Supervisor about the plan of FYP II

3. PROBLEMS ENCOUNTERED

- N/A

4. SELF EVALUATION OF THE PROGRESS

- Speed more time to review back the FYP I

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 4**
Student Name & ID: Chia Zhen Yik & 19ACB04782	
Supervisor: Dr. Aun Yichiet	
Project Title: **TELCO RECOMMENDATION BASED ON SPEED TEST DATA**	

1. WORK DONE

- Design a haversine algorithm.

2. WORK TO BE DONE

- Retrieve geographical coordinate of user.
- Host a speed test system.

3. PROBLEMS ENCOUNTERED

- Laptop don't have own GPS chip, it can't get the accurate coordinate of user.

4. SELF EVALUATION OF THE PROGRESS

- Research on how to get accurate geographical coordinate from user.
- Research on the Download Speed, Upload Speed, Jiiter&Ping Algorithm.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 6**
Student Name & ID: Chia Zhen Yik & 19ACB04782	
Supervisor: Dr. Aun Yichiet	
Project Title: **TELCO RECOMMENDATION BASED ON SPEED TEST DATA**	

1. WORK DONE

- Host a speed test system.

2. WORK TO BE DONE

- Build a database to store the speed test result
- Build a phpMyAdmin to manage the database.
- Provide Test ID for each result.

3. PROBLEMS ENCOUNTERED

- Connectivity problem for the speed test website and the database

4. SELF EVALUATION OF THE PROGRESS

- Research on each database in Ubuntu

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 8**
Student Name & ID: Chia Zhen Yik & 19ACB04782	
Supervisor: Dr. Aun Yichiet	
Project Title: **TELCO RECOMMENDATION BASED ON SPEED TEST DATA**	

1. WORK DONE

- Able to get the geographical coordinate of user in laptop.
- Created Database for the speed test website

2. WORK TO BE DONE

- Provide Test ID for each result.

3. PROBLEMS ENCOUNTERED

- It is difficult to let each data have own unique Test ID

4. SELF EVALUATION OF THE PROGRESS

- Explore more method to get the current user' geographical coordinate

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 10**
Student Name & ID: Chia Zhen Yik & 19ACB04782	
Supervisor: Dr. Aun Yichiet	
Project Title: **TELCO RECOMMENDATION BASED ON SPEED TEST DATA**	

1. WORK DONE

- Each Speed test result have their own unique Test ID

2. WORK TO BE DONE

- Visualize the Data on the google map

3. PROBLEMS ENCOUNTERED

- Python can't provide a powerful data visualization map.

4. SELF EVALUATION OF THE PROGRESS

- Must spend more time to learn how to visualize the data on the map.

Supervisor's signature

Student's signature

FINAL YEAR PROJECT WEEKLY REPORT

(Project II)

Trimester, Year: Y3S3	**Study week no.: 12**
Student Name & ID: Chia Zhen Yik & 19ACB04782	
Supervisor: Dr. Aun Yichiet	
Project Title: **TELCO RECOMMENDATION BASED ON SPEED TEST DATA**	

1. WORK DONE

- Visualize the data on the google map

2. WORK TO BE DONE

- Optimize the visualize map.

3. PROBLEMS ENCOUNTERED

- The progress is smooth.

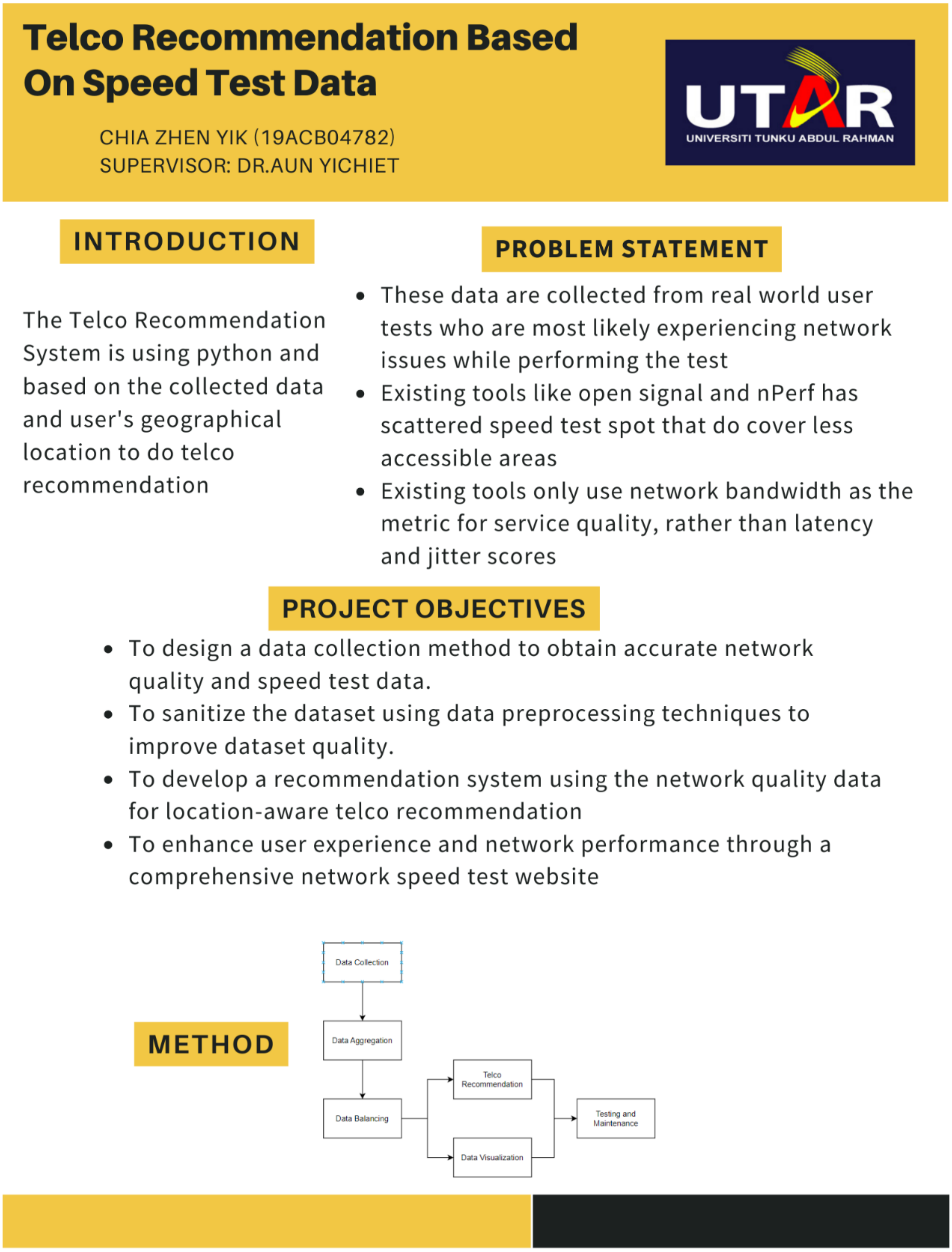
4. SELF EVALUATION OF THE PROGRESS

- Should spend more time on the research for the visualize data on the map

Supervisor's signature

Student's signature

A.4 POSTER



Telco Recommendation Based On Speed Test Data

CHIA ZHEN YIK (19ACB04782)
SUPERVISOR: DR.AUN YICHIET

UTAR
UNIVERSITI TUNKU ABDUL RAHMAN

INTRODUCTION

The Telco Recommendation System is using python and based on the collected data and user's geographical location to do telco recommendation

PROBLEM STATEMENT

- These data are collected from real world user tests who are most likely experiencing network issues while performing the test
- Existing tools like open signal and nPerf has scattered speed test spot that do cover less accessible areas
- Existing tools only use network bandwidth as the metric for service quality, rather than latency and jitter scores

PROJECT OBJECTIVES

- To design a data collection method to obtain accurate network quality and speed test data.
- To sanitize the dataset using data preprocessing techniques to improve dataset quality.
- To develop a recommendation system using the network quality data for location-aware telco recommendation
- To enhance user experience and network performance through a comprehensive network speed test website

METHOD

Data Collection

Data Aggregation

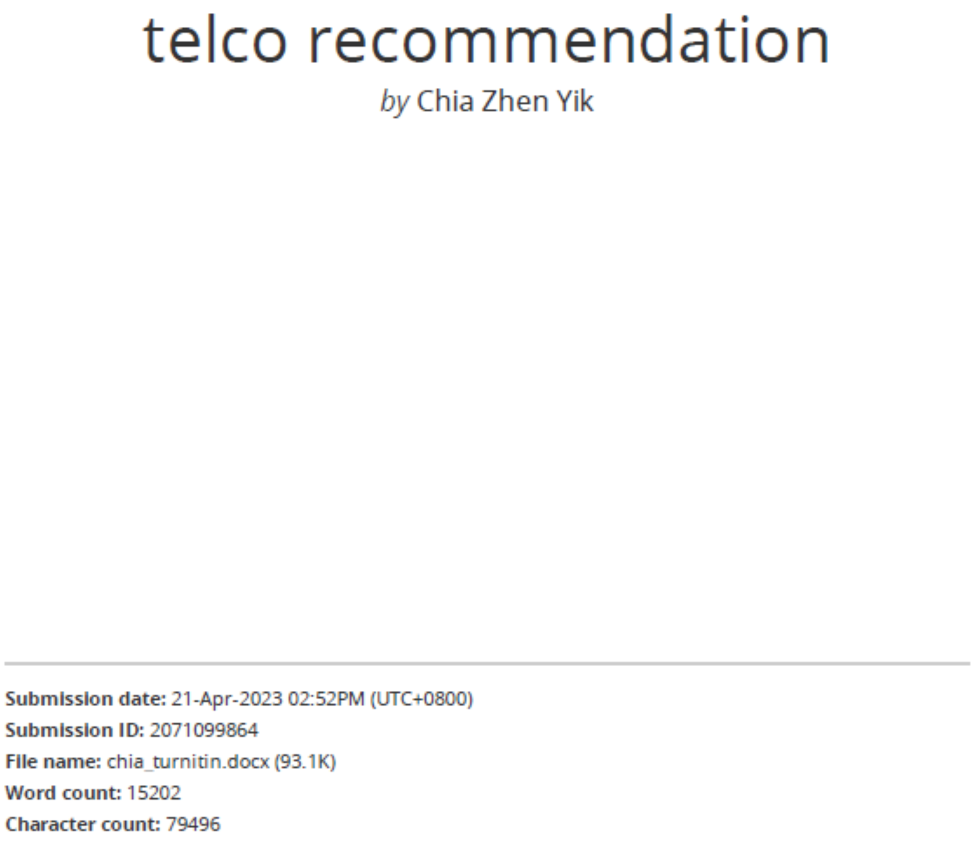
Data Balancing

Telco Recommendation

Data Visualization

Testing and Maintenance

PLAGIARISM CHECK RESULT

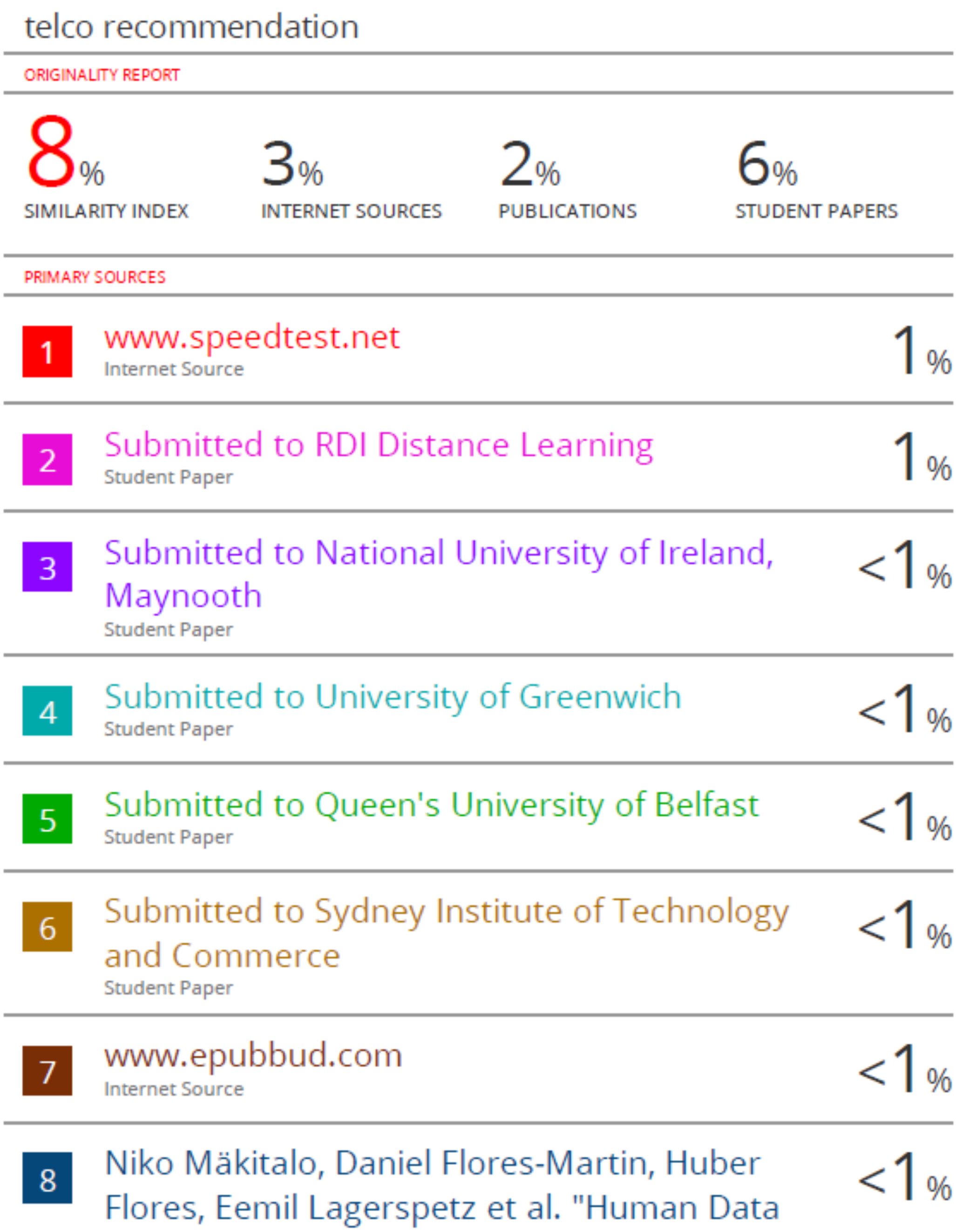


telco recommendation

by Chia Zhen Yik

Submission date: 21-Apr-2023 02:52PM (UTC+0800)
Submission ID: 2071099864
File name: chia_turnitin.docx (93.1K)
Word count: 15202
Character count: 79496

PLAGIARISM CHECK RESULT



telco recommendation

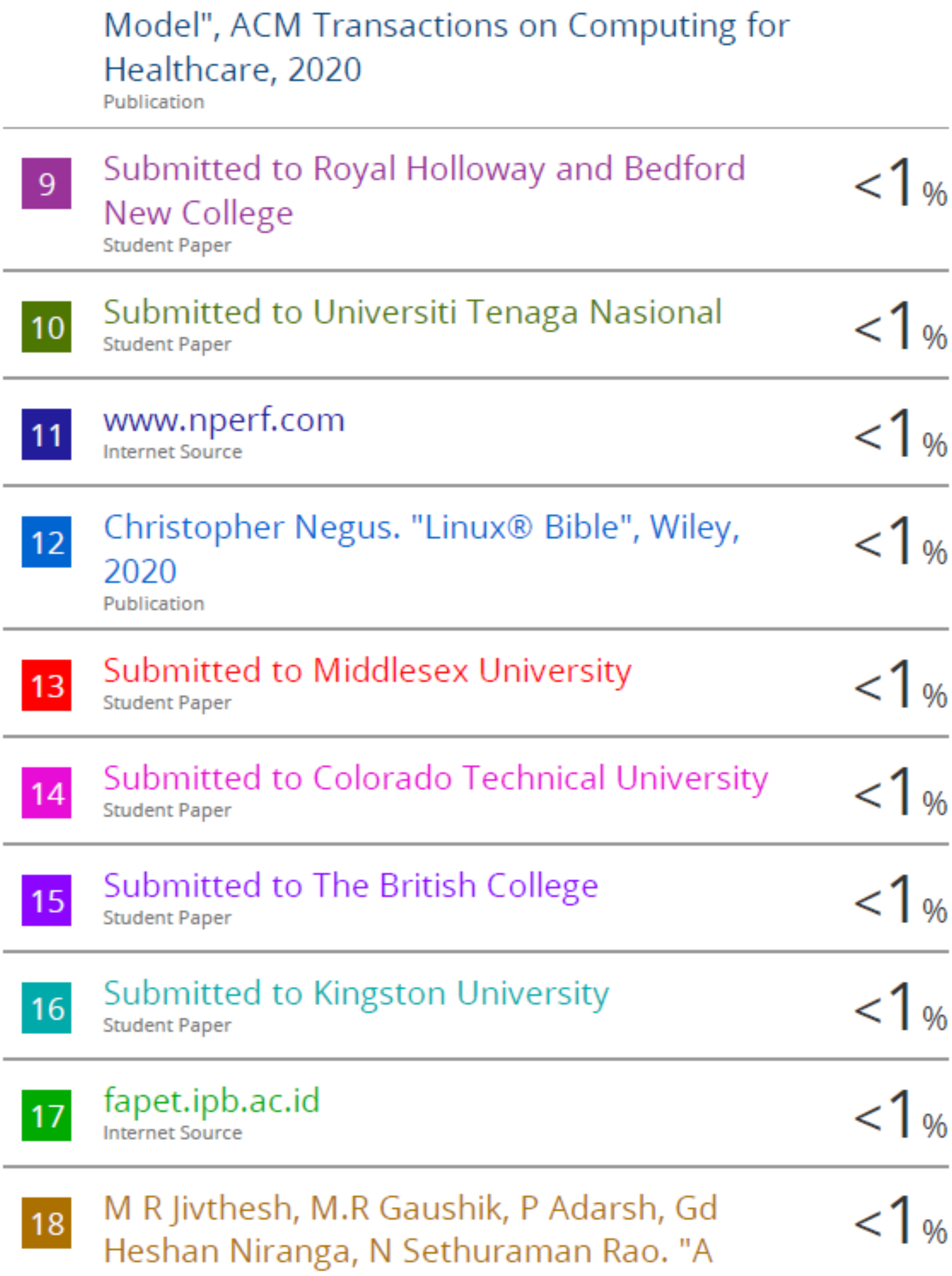
ORIGINALITY REPORT

8%	3%	2%	6%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

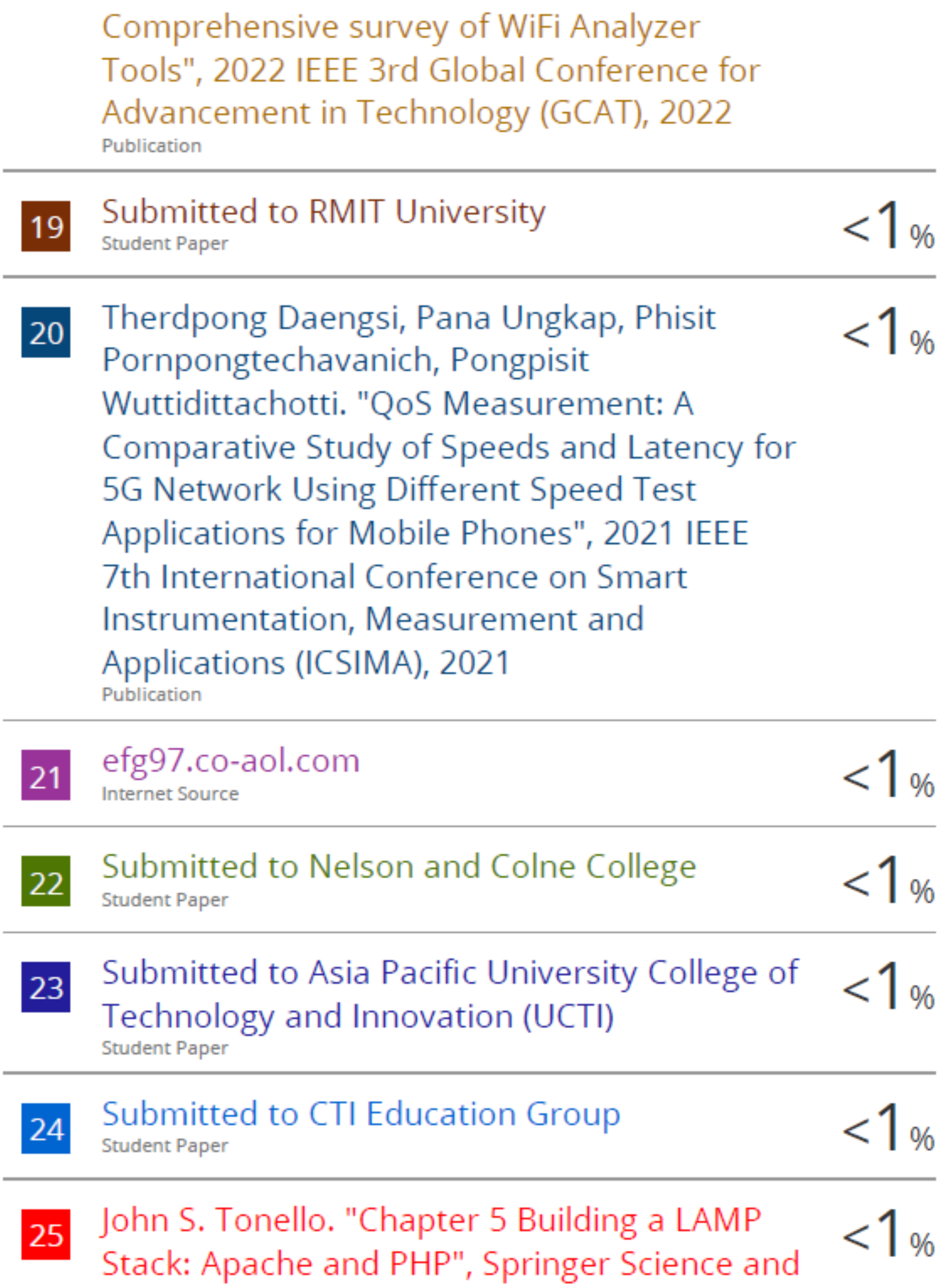
PRIMARY SOURCES

1	www.speedtest.net Internet Source	1%
2	Submitted to RDI Distance Learning Student Paper	1%
3	Submitted to National University of Ireland, Maynooth Student Paper	<1%
4	Submitted to University of Greenwich Student Paper	<1%
5	Submitted to Queen's University of Belfast Student Paper	<1%
6	Submitted to Sydney Institute of Technology and Commerce Student Paper	<1%
7	www.epubbud.com Internet Source	<1%
8	Niko Mäkitalo, Daniel Flores-Martin, Huber Flores, Eemil Lagerspetz et al. "Human Data	<1%

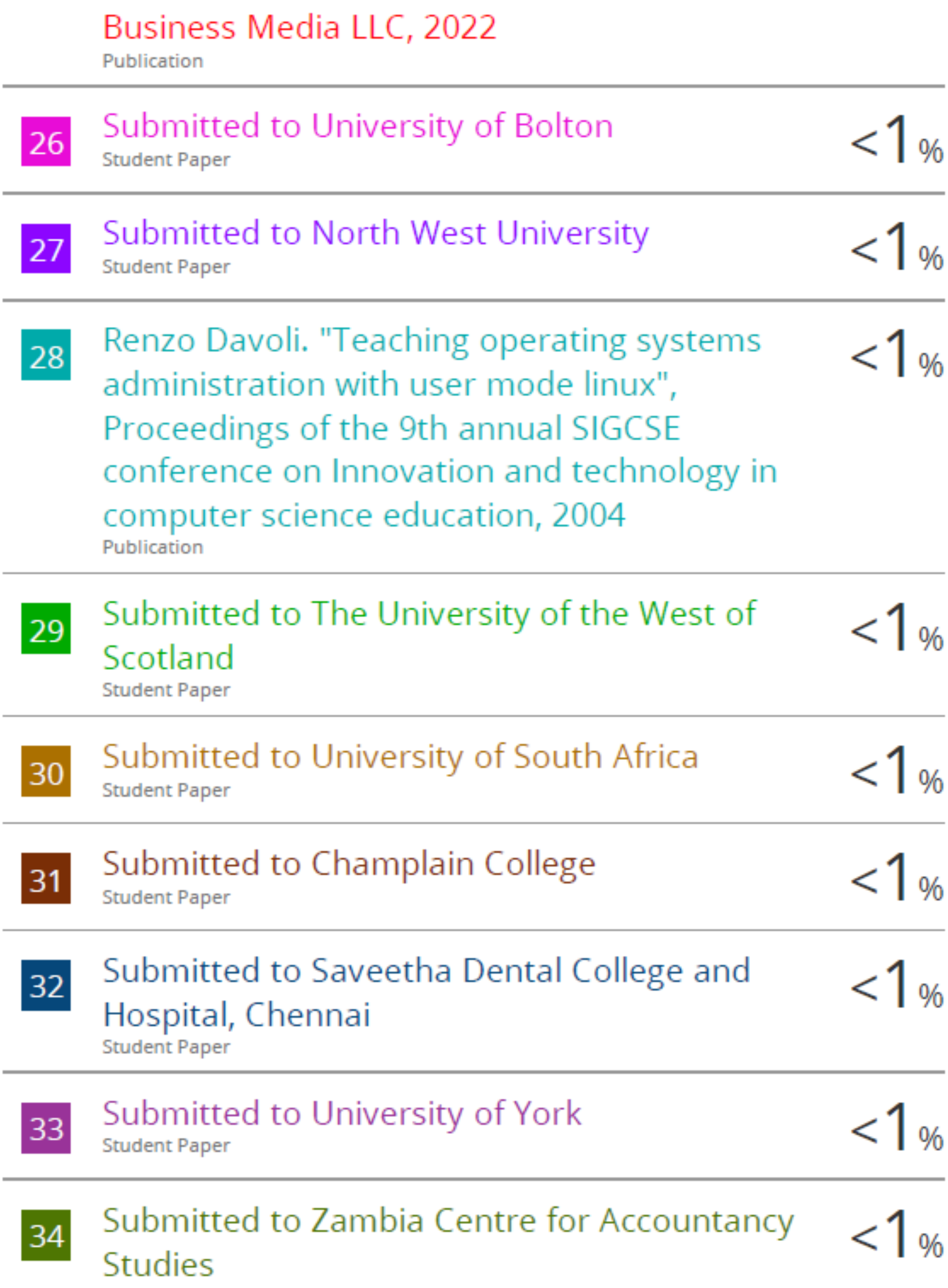
PLAGIARISM CHECK RESULT



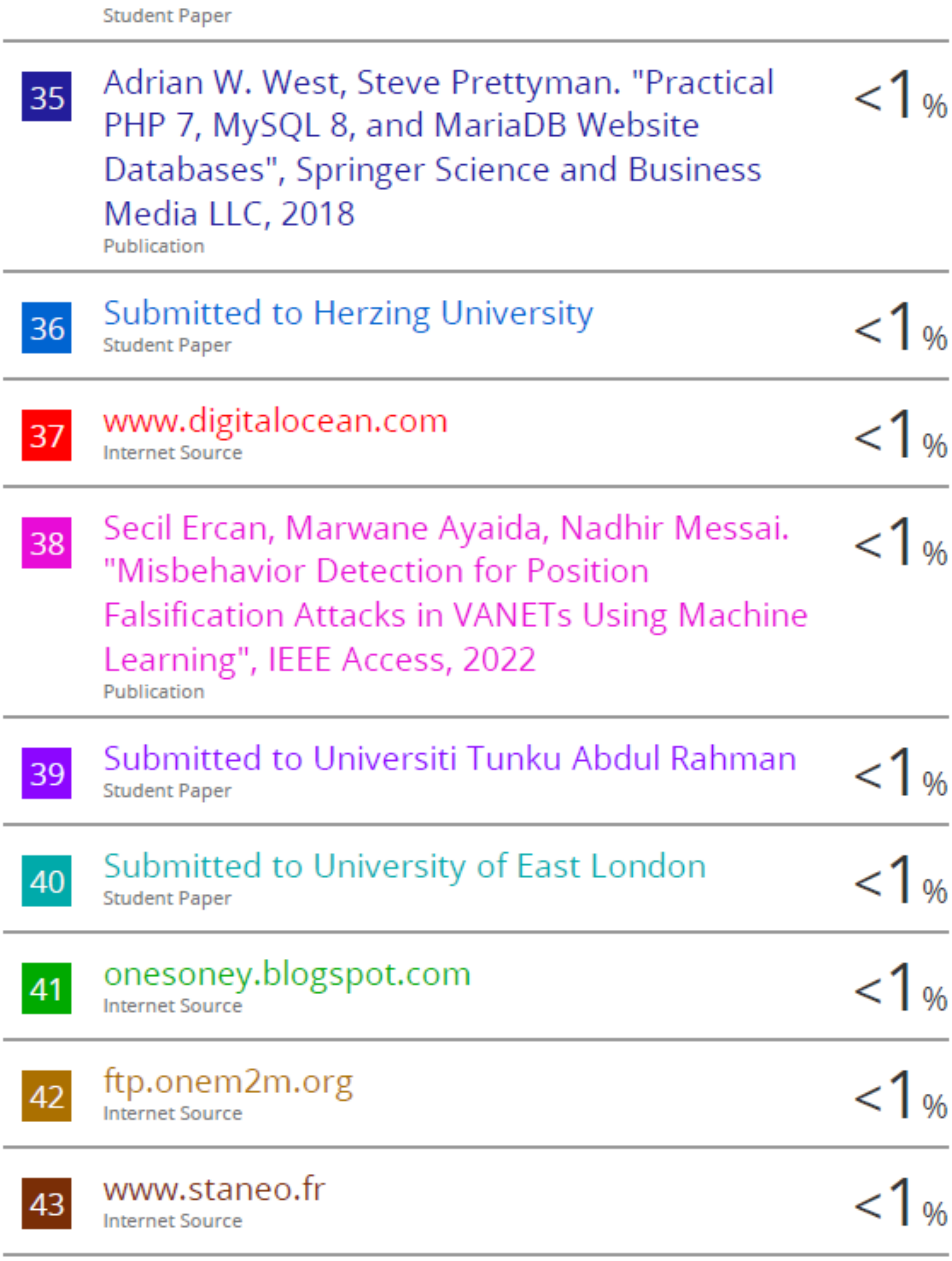
	Model", ACM Transactions on Computing for Healthcare, 2020 Publication	
9	Submitted to Royal Holloway and Bedford New College Student Paper	<1 %
10	Submitted to Universiti Tenaga Nasional Student Paper	<1 %
11	www.nperf.com Internet Source	<1 %
12	Christopher Negus. "Linux® Bible", Wiley, 2020 Publication	<1 %
13	Submitted to Middlesex University Student Paper	<1 %
14	Submitted to Colorado Technical University Student Paper	<1 %
15	Submitted to The British College Student Paper	<1 %
16	Submitted to Kingston University Student Paper	<1 %
17	fapet.ipb.ac.id Internet Source	<1 %
18	M R Jivthesh, M.R Gaushik, P Adarsh, Gd Heshan Niranga, N Sethuraman Rao. "A	<1 %



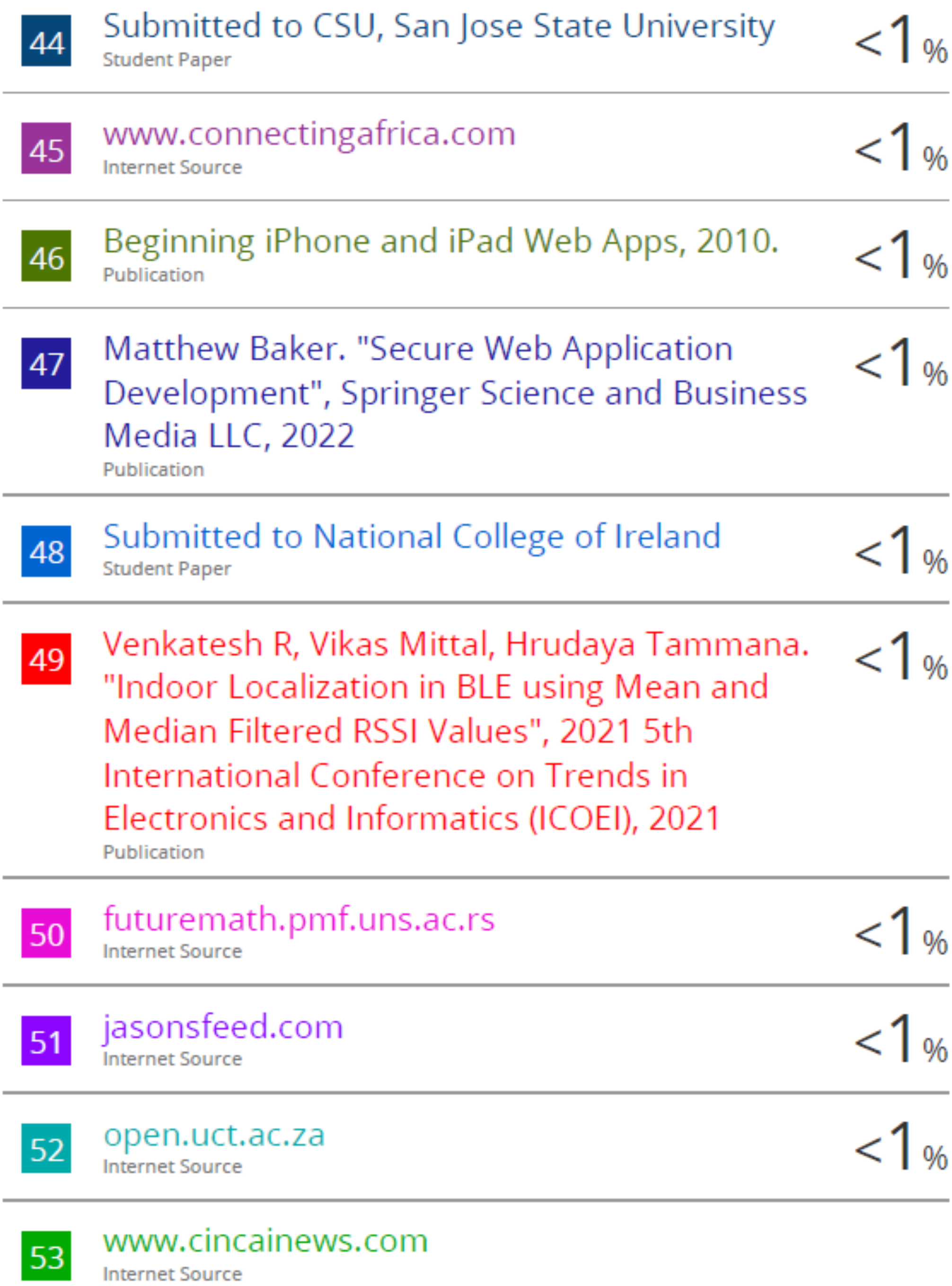
	Comprehensive survey of WiFi Analyzer Tools", 2022 IEEE 3rd Global Conference for Advancement in Technology (GCAT), 2022 Publication	
19	Submitted to RMIT University Student Paper	<1 %
20	Therdpong Daengsi, Pana Ungkap, Phisit Pornpongtechavanich, Pongpisit Wuttidittachotti. "QoS Measurement: A Comparative Study of Speeds and Latency for 5G Network Using Different Speed Test Applications for Mobile Phones", 2021 IEEE 7th International Conference on Smart Instrumentation, Measurement and Applications (ICSIMA), 2021 Publication	<1 %
21	efg97.co-aol.com Internet Source	<1 %
22	Submitted to Nelson and Colne College Student Paper	<1 %
23	Submitted to Asia Pacific University College of Technology and Innovation (UCTI) Student Paper	<1 %
24	Submitted to CTI Education Group Student Paper	<1 %
25	John S. Tonello. "Chapter 5 Building a LAMP Stack: Apache and PHP", Springer Science and	<1 %



	Business Media LLC, 2022 Publication	
26	Submitted to University of Bolton Student Paper	<1 %
27	Submitted to North West University Student Paper	<1 %
28	Renzo Davoli. "Teaching operating systems administration with user mode linux", Proceedings of the 9th annual SIGCSE conference on Innovation and technology in computer science education, 2004 Publication	<1 %
29	Submitted to The University of the West of Scotland Student Paper	<1 %
30	Submitted to University of South Africa Student Paper	<1 %
31	Submitted to Champlain College Student Paper	<1 %
32	Submitted to Saveetha Dental College and Hospital, Chennai Student Paper	<1 %
33	Submitted to University of York Student Paper	<1 %
34	Submitted to Zambia Centre for Accountancy Studies	<1 %



	Student Paper	
35	Adrian W. West, Steve Prettyman. "Practical PHP 7, MySQL 8, and MariaDB Website Databases", Springer Science and Business Media LLC, 2018 Publication	<1 %
36	Submitted to Herzing University Student Paper	<1 %
37	www.digitalocean.com Internet Source	<1 %
38	Secil Ercan, Marwane Ayaida, Nadhir Messai. "Misbehavior Detection for Position Falsification Attacks in VANETs Using Machine Learning", IEEE Access, 2022 Publication	<1 %
39	Submitted to Universiti Tunku Abdul Rahman Student Paper	<1 %
40	Submitted to University of East London Student Paper	<1 %
41	onesoney.blogspot.com Internet Source	<1 %
42	ftp.onem2m.org Internet Source	<1 %
43	www.staneo.fr Internet Source	<1 %



44	Submitted to CSU, San Jose State University Student Paper	<1 %
45	www.connectingafrica.com Internet Source	<1 %
46	Beginning iPhone and iPad Web Apps, 2010. Publication	<1 %
47	Matthew Baker. "Secure Web Application Development", Springer Science and Business Media LLC, 2022 Publication	<1 %
48	Submitted to National College of Ireland Student Paper	<1 %
49	Venkatesh R, Vikas Mittal, Hrudaya Tammana. "Indoor Localization in BLE using Mean and Median Filtered RSSI Values", 2021 5th International Conference on Trends in Electronics and Informatics (ICOEI), 2021 Publication	<1 %
50	futuremath.pmf.uns.ac.rs Internet Source	<1 %
51	jasonsfeed.com Internet Source	<1 %
52	open.uct.ac.za Internet Source	<1 %
53	www.cincainews.com Internet Source	

PLAGIARISM CHECK RESULT

Form Title: Supervisor's Comments on Originality Report Generated by Turnitin for Submission of Final Year Project Report (for Undergraduate Programmes)			
Form Number: FM-IAD-005	Rev No.: 0	Effective Date: 01/10/2013	Page No.: 1of 1

FACULTY OF INFORMATION AND COMMUNICATION TECHNOLOGY

Full Name(s) of Candidate(s)	CHIA ZHEN YIK
ID Number(s)	19ACB04782
Programme / Course	CN
Title of Final Year Project	TELCO RECOMMENDATION BASED ON SPEED TEST DATA

Similarity	**Supervisor's Comments (Compulsory if parameters of originality exceed the limits approved by UTAR)**
Overall similarity index: 8 **%** **Similarity by source** Internet Sources: 3 % Publications: 2 % Student Papers: 6 %	
Number of individual sources listed of more than 3% similarity: 0	

Parameters of originality required, and limits approved by UTAR are as Follows:

(i) Overall similarity index is 20% and below, and
(ii) Matching of individual sources listed must be less than 3% each, and
(iii) Matching texts in continuous block must not exceed 8 words

Note: Parameters (i) – (ii) shall exclude quotes, bibliography and text matches which are less than 8 words.

Note: Supervisor/Candidate(s) is/are required to provide softcopy of full set of the originality report to Faculty/Institute

Based on the above results, I hereby declare that I am satisfied with the originality of the Final Year Project Report submitted by my student(s) as named above.

Signature of Supervisor

Name: Aun Yichiet

Date: 24/4/2023

Signature of Co-Supervisor

Name: ______

Date: ______

FYP 2 CHECKLIST

UNIVERSITI TUNKU ABDUL RAHMAN

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY (KAMPAR CAMPUS)

CHECKLIST FOR FYP2 THESIS SUBMISSION

Student Id	19ACB04782
Student Name	CHIA ZHEN YIK
Supervisor Name	Dr. AUN YICHIET

TICK (√)	**DOCUMENT ITEMS** Your report must include all the items below. Put a tick on the left column after you have checked your report with respect to the corresponding item.
√	Front Plastic Cover (for hardcopy)
√	Title Page
√	Signed Report Status Declaration Form
√	Signed FYP Thesis Submission Form
√	Signed form of the Declaration of Originality
√	Acknowledgement
√	Abstract
√	Table of Contents
√	List of Figures (if applicable)
√	List of Tables (if applicable)
√	List of Symbols (if applicable)
√	List of Abbreviations (if applicable)
√	Chapters / Content
√	Bibliography (or References)
√	All references in bibliography are cited in the thesis, especially in the chapter of literature review
√	Appendices (if applicable)
√	Weekly Log
√	Poster
√	Signed Turnitin Report (Plagiarism Check Result - Form Number: FM-IAD-005)
√	I agree 5 marks will be deducted due to incorrect format, declare wrongly the ticked of these items, and/or any dispute happening for these items in this report.

*Include this form (checklist) in the thesis (Bind together as the last page)

I, the author, have checked and confirmed all the items listed in the table are included in my report.

____

(Signature of Student)
Date: 21/4/2023