

**Download simultaneously using multiple Internet Connections**

By

Arief Fikhrie bin Mat Yamin

Dissertation submitted in partial fulfillment of  
requirements for the  
Bachelor of Technology (Hons)  
(Information & Communication Technology)

MAY 2013

Universiti Teknologi PETRONAS  
Bandar Seri Iskandar,  
31750 Tronoh  
Perak Darul Ridzuan

## **CERTIFICATION OF ORIGINALITY**

This is to certify that I am responsible for the work submitted in this project, that the original work is my own except as specified in the references and acknowledgements, and that the original work contained herein have not been undertaken or done by unspecified sources or persons.

---

Arief Fikhrie bin Mat Yamin

## ABSTRACT

This report discusses the research done on the selected topic, which is **downloading simultaneously using multiple internet connections**. This project shows that multiple internet connection can be used to download file simultaneously. The main idea for this project is to increase the internet speed to download a file and to optimize the usage of multiple internet connection. This report covers the scope of networking, file system and Java programming. There are internet users that have more than one internet connection such as broadband and home internet. The purpose of this project was to leverage the use of multiple internet connection a user has in order to increase the speed of download. Tools that are going to be used are network interface device such as wireless adapter, broadband dongle and Ethernet adapter. Besides, Netbeans was used to develop the prototype. This project methodology is using agile programming method as the main method in developing the user interface of the prototype. As for the result, this system successfully reduces download time to download a file by using multiple internet connection.

## Table of Contents

CERTIFICATION OF ORIGINALITY .....	ii
ABSTRACT	iii
Table of Contents .....	iv
Table of Figures .....	vi
Chapter 1: Introduction.....	1
1.1 Background .....	1
1.1.1 Internet Connection.....	2
1.1.2 Internet Service Provider .....	3
1.1.3 Data Transfer .....	4
1.2 Problem Statement .....	5
1.3 Objectives.....	6
Chapter 2: Literature Review and Theory .....	6
2.1 Leveraging multiple internet connection by load balancing the bandwidth .....	6
2.2 Speed of internet connection around the world.....	7
2.3 Network throughput in normal environment.....	8
2.4 Network latency in downloading files from internet. ....	9
2.5 File chunking is being used to improve the response time of download .....	10
2.6 Binding application to a specific interface.....	12
2.7 Transmission Control Protocol and User Datagram Protocol.....	12

2.8	Download simultaneously using multiple internet connection theory .....	13
Chapter 3: Methodology .....		15
3.1	Research Methodology.....	15
3.1.1	Planning .....	16
3.1.2	Analysis.....	17
3.1.3	Design .....	20
3.1.4	Development .....	23
3.1.5	Implementation .....	32
3.2	Tool Used .....	35
Chapter 4: Result and Discussion.....		37
4.1	Several factors need to take measure before making decision.....	37
4.2	Data result .....	38
4.3	Multiple default gateway can cause connectivity problems.....	39
4.4	Reduce downloading times .....	40
4.5	Downloading time become slower .....	41
Chapter 5: Conclusion .....		42
Chapter 6: Reference .....		43
APPENDICES .....		45

## Table of Figures

Figure 1: Shopping needs online.....	1
Figure 2: IPTV allows watching online by using internet connection.....	2
Figure 3: A visualization of routing paths through a portion of the Internet .....	2
Figure 4: ARPANET, initial research of internet .....	3
Figure 5: Multiple Accesses Conceptual Model.....	7
Figure 6: Internet speed around the world .....	8
Figure 7: Example of iPerf throughput testing .....	9
Figure 8: Method used to calculate network latency .....	10
Figure 9: Client server architecture.....	11
Figure 10: Difference between TCP and UDP .....	13
Figure 11: Cycle of Project Phases .....	15
Figure 12: Creating proxy for Celcom Broadband in Virtual Machine.....	18
Figure 13: Response from HTTP Server .....	19
Figure 14: Flow chart of the system .....	21
Figure 15: Using loop to merge the chunked file .....	22
Figure 16: Class Diagram for download manager .....	23
Figure 17: Simple interface.....	29
Figure 18: Main Download Interface.....	29
Figure 19: URL input.....	29
Figure 20: Download Information .....	30
Figure 21: Download progress.....	30
Figure 22: The executable program for this project.....	32

Figure 23: Download sample file from internet.....	33
Figure 24: Download Information .....	33
Figure 25: Download Progress.....	33
Figure 26: Successfully download file.....	34
Figure 27: Download file using fire.....	34
Figure 28: Left side - file download using this project downloader, right side - file download using Firefox .....	35
Figure 29: Show the file is identical .....	35
Figure 30: Network interface 1 detected as dead by TCP .....	39
Figure 31: Download using 1 Internet Connection - Celcom Broadband .....	40
Figure 32: Download using 2 Internet Connection - Celcom Broadband and UTP LAN .....	41

# Chapter 1: Introduction

## 1.1 Background

Nowadays, we live in the world that constantly developing and sometimes we may find it hard to follow up with current trend. Internet is one of the trends that we already consider it as necessity to our daily lives which allows us to get recent news, communicate with people and even to pay bills just using one click.

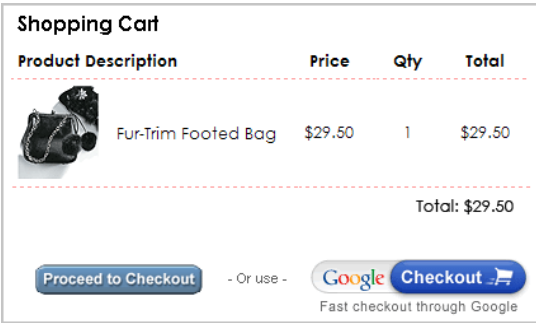


Figure 1: Shopping needs online

As this trend is wide to all people around the world, peoples can share their document or music files using free webhosting in World Wide Web and allow people to download it and save it in their computer. There are many companies provide service for internet connection, as the people demand more bandwidth limitation and faster rate of internet speed the price of the internet service become much expensive.





Figure 2: IPTV allows watching online by using internet connection

### 1.1.1 Internet Connection

Internet is network connections that connect all computer users around the world by using standard Internet protocol suite (TCP/IP). It consists of many networks from the home to big organization users that connected on a gateways or routers and make them linked together. Internet is needed to carry information and users can view the information using an internet browser. Internet access is the means by which individual terminals, computers, mobile devices, and local area networks are connected to the global Internet. Basically, it is a source through which Internet users can access Internet services.

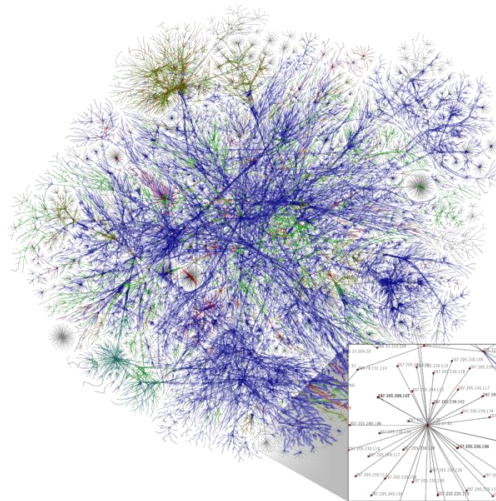


Figure 3: A visualization of routing paths through a portion of the Internet

Internet access is usually sold by Internet Service Providers (ISPs) that use many different technologies offering a wide range of data rates to the end user. Internet is originating from U.S. Defense Advanced Research Projects Agency (DARPA) that initiated research on packet switching in 1960s. That research has resulted into ARPANET which became a protocol for internetworking that allows separate network to be connected together into bigger networks. In early 1990s, commercialization of internet started and now over a third of world's human population are using Internet services.

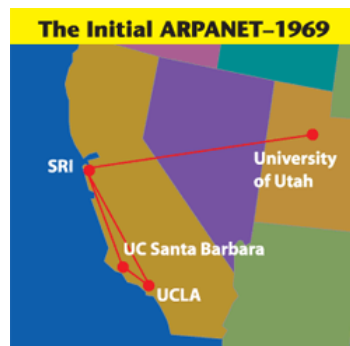


Figure 4: ARPANET, initial research of internet

### 1.1.2 Internet Service Provider

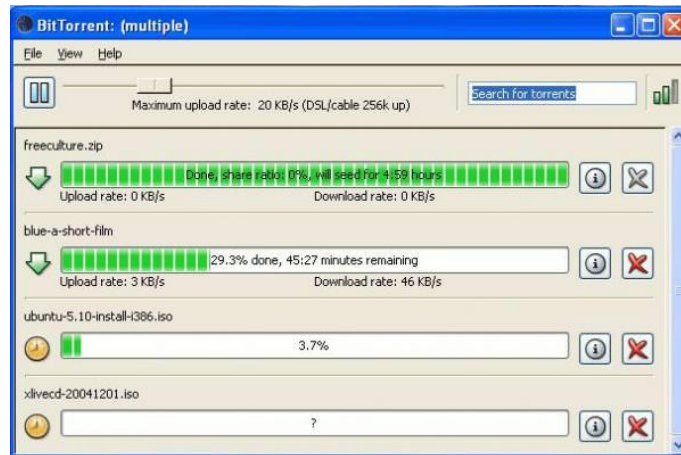
An Internet Service Provider (ISP) is organizations that provide people an access to Internet. They provide services to get access into internet by connecting user by via copper wires, fiber optic or wireless connections. ISP can be classified into few classifications which are Access Provider, Mailbox Provider, Hosting ISP, Transit ISP and Virtual ISP.

In Malaysia, Telekom Malaysia has dominated the Internet Access Provider for fixed cable line while broadband internet access is provided by Celcom, Maxis, Digi, 4G YES and P1MAX. People need to consider subscribing to internet connection as service provider got their own specification of services that mostly based on internet speed, bandwidth and price.

### **1.1.3 Data Transfer**

As the role of the Internet becomes more and more varied, Internet and multimedia cannot be thought of separately any more. There may be the common end users who get information provided in the form of multimedia and there also may be the multimedia developers who are delivering their products across the Internet. In both cases, one of the critical factors in determining how enjoyable and effective their internet experience will be how much time will be spent for uploading or downloading which is internet access speed. To have fast Internet Access speed is depending on which ISP one has to choose and the country's current technology.

File sharing is one of the ways to transfer large amount of data via the internet. Files can be sent through email, share through website and peer to peer network. Any files can be uploaded into webserver and able to be viewed by providing IP address or domain name to internet user that we wish to share.



Nowadays, files are uploaded into webserver as a public download. Downloading public file are risks that need to be incurred by internet user itself which to avoid a malicious software or computer virus that can give negative effect on their computer or user themselves. Some people that want to share their files might want to upload their files to secure websites and to download such files internet users need to login into the secure websites which web pages invisible to public and then began to download the private file.

## 1.2 Problem Statement

In Malaysia, there are several Internet service providers that allowed multiple internet connection accesses at a time. However we are unable to use all of the internet accesses such as wireless broadband and home internet at once. This limitation of internet access makes people perceive that a single internet connection sufficient for internet connections thus blame the ISPs when the speed of internet connection is slow. Furthermore, most people do not realize that they have multiple internet connection access. This study, there are three main problems that as below:

1. How to download multiple internet files at the same time by using multiple internet connections?
2. How do internet users optimize the usage of multiple internet connection?
3. How to increase speed of download?

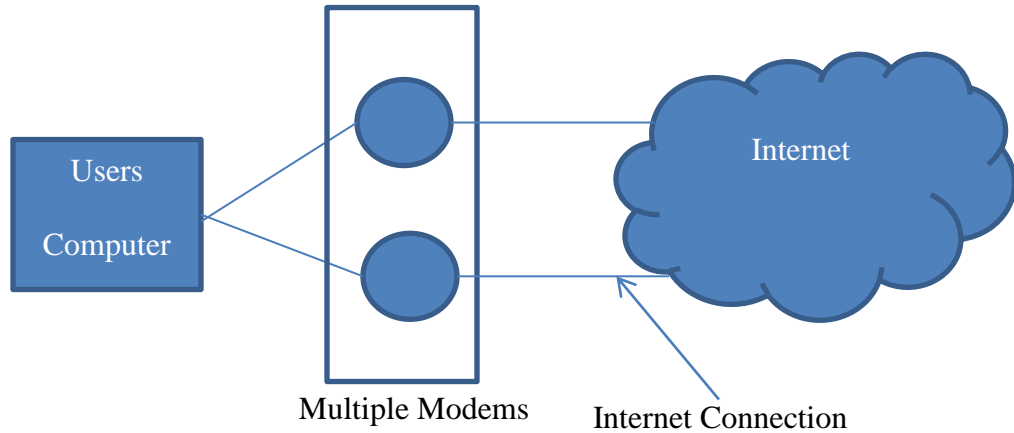
### **1.3 Objectives**

- To optimize the usage of multiple internet connections in a computer
- Increasing internet file download speed in a computer

## **Chapter 2: Literature Review and Theory**

### **2.1 Leveraging multiple internet connection by load balancing the bandwidth**

The increasing trend of file sharing in internet has elevated the usage of internet access bandwidth. Thus any internet users are seeking ways to increase the internet access bandwidth since the bandwidth on a single internet access is very much limited. By combining multiple internet connections, internet users can accumulate enough bandwidth to serve their needs (Vicomsoft, 2012).



**Figure 5: Multiple Accesses Conceptual Model**

Figure 1 shows a basic model with multiple modems to aggregate bandwidth of the internet connections.

## **2.2 Speed of internet connection around the world**

According to Daily info graphic Japan is in the first place for the fastest internet speed and it is being ranked by ITIF (Information Technology & Innovation Foundation, 2011). Even though Malaysia did not earn the rank of below 100, there is still various choice of internet being offered in the country.

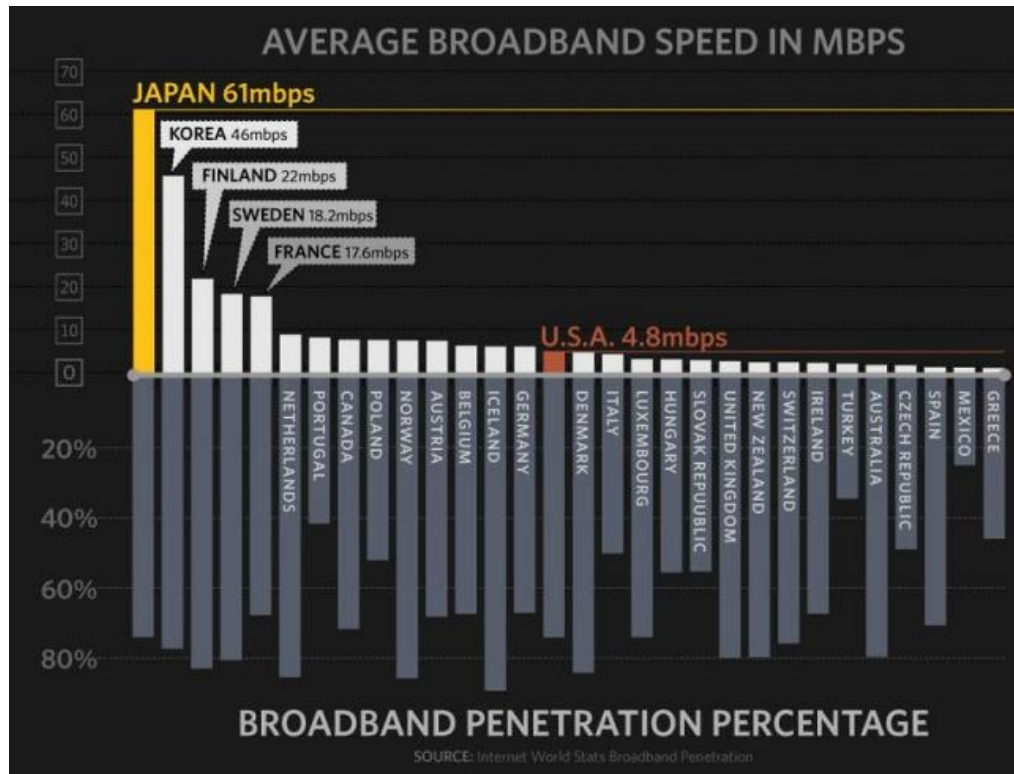


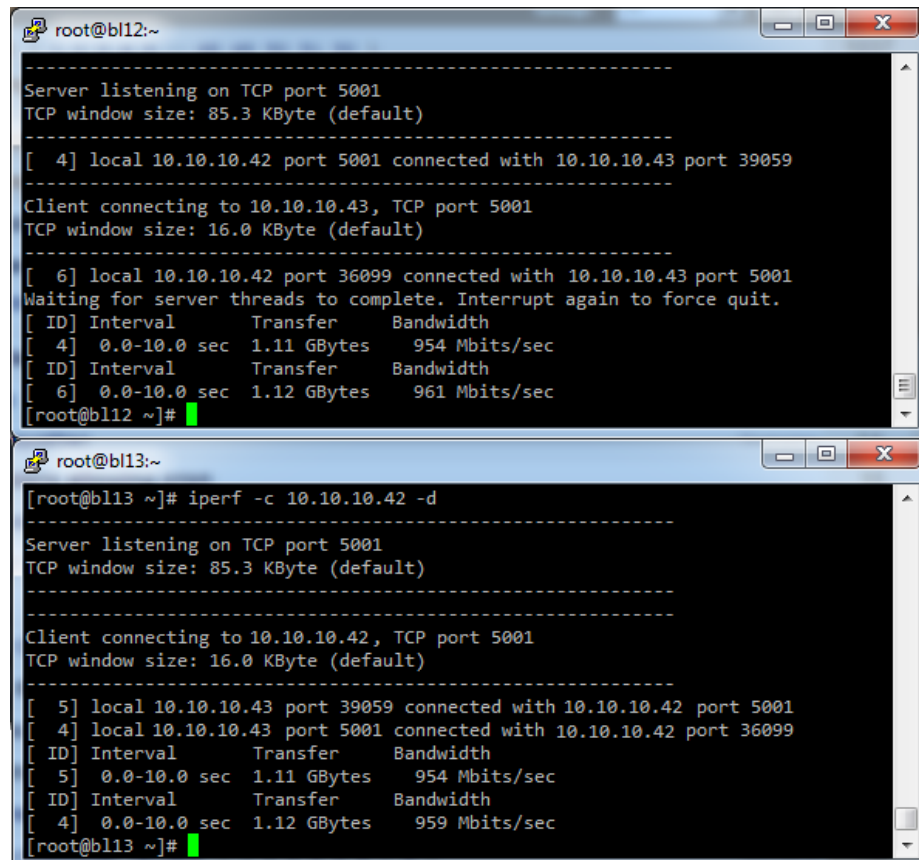
Figure 6: Internet speed around the world

The speed of the internet offered varies according to the cost that subscribers are willing to pay. The internet speed could become faster if all the internet connections that a user subscribes to can be combined.

### 2.3 Network throughput in normal environment

Network performance is measured by the amount of file data that can be transferred over a channel at a time. Throughput of a system can be measured by various tools such as iPerf. In a testing environment of internet connection device, the testing engineers will be using iPerf to test the network performance (Tirumala, Cottrell, & Dunigan, 2003). The testing conducted by executing iPerf in both client and server and the client will send a generated packet to the server. The result of the network is then analyzed by viewing the log files whether or not

the size of packet transferred from client is same with the amount of packet received at server side. The result will be logged according to an interval of time and the network is said to be in a good performance if the packets sent and received has the lesser or same amount.



```
root@bl12:~
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
[ 4] local 10.10.10.42 port 5001 connected with 10.10.10.43 port 39059
-----
Client connecting to 10.10.10.43, TCP port 5001
TCP window size: 16.0 KByte (default)
-----
[ 6] local 10.10.10.42 port 36099 connected with 10.10.10.43 port 5001
Waiting for server threads to complete. Interrupt again to force quit.
[ ID] Interval      Transfer    Bandwidth
[ 4] 0.0-10.0 sec  1.11 GBytes  954 Mbits/sec
[ ID] Interval      Transfer    Bandwidth
[ 6] 0.0-10.0 sec  1.12 GBytes  961 Mbits/sec
[root@bl12 ~]#

root@bl13:~
[root@bl13 ~]# iperf -c 10.10.10.42 -d
-----
Server listening on TCP port 5001
TCP window size: 85.3 KByte (default)
-----
Client connecting to 10.10.10.42, TCP port 5001
TCP window size: 16.0 KByte (default)
-----
[ 5] local 10.10.10.43 port 39059 connected with 10.10.10.42 port 5001
[ 4] local 10.10.10.43 port 5001 connected with 10.10.10.42 port 36099
[ ID] Interval      Transfer    Bandwidth
[ 5] 0.0-10.0 sec  1.11 GBytes  954 Mbits/sec
[ ID] Interval      Transfer    Bandwidth
[ 4] 0.0-10.0 sec  1.12 GBytes  959 Mbits/sec
[root@bl13 ~]#
```

Figure 7: Example of iPerf throughput testing

## 2.4 Network latency in downloading files from internet.

The method or system to measure network latency in peer-to-peer architecture has been patented in the United States. This system can also be adapted into non peer-to-peer environment because the concept applied is the same by timestamping packet. Each client in the peer-to-peer architecture determines the network latency



via a ping response and protocol response (A., Link, Seattle, Im, & Bellevue, 1998). The first client will timestamp the into a (ping) data packet and sends the packet to second client, who places the information into the packet and sends the packet as a response back to the first client.

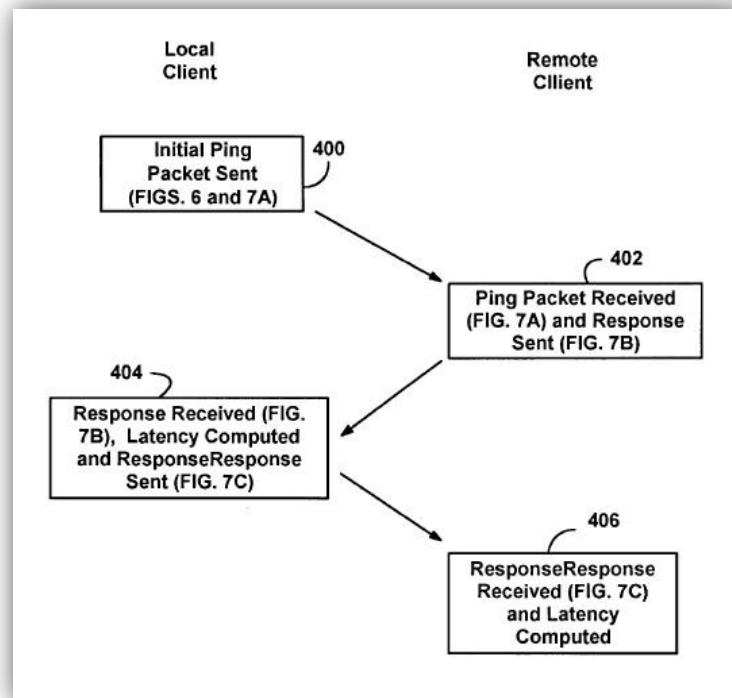
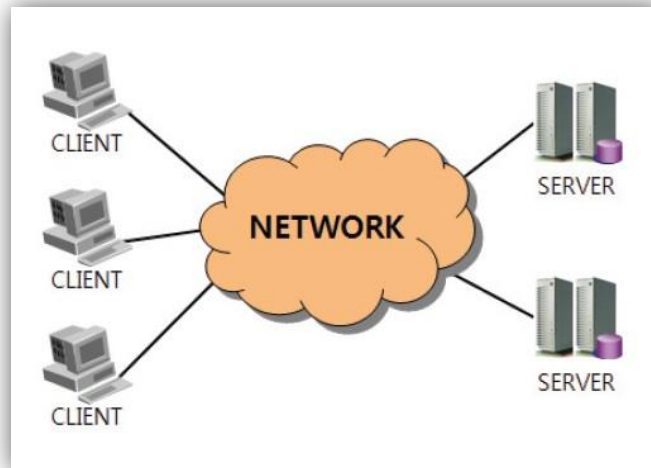


Figure 8: Method used to calculate network latency

## 2.5 File chunking is being used to improve the response time of download

Chunking is a specific feature of the HTTP 1.1 protocol. That refers to a facility that allows inconveniently large messages to be broken into conveniently-sized smaller "chunks." File chunking has been used in backing up massive data storage based on a client and server architecture. The process of chopping down the size of data called the "chunking" happened at the client (Youjip Won, 2008). There are 2 types of data chunking which are fixed size chunking and variable size

chunking. Fixed size chunking simply means if the file is going to be divided in 8Kilobytes, therefore other chunked parts should have the same size too. These two types of chunking were being used in process of backing up massive data.



**Figure 9: Client server architecture**

Fixed size chunking has its own drawbacks and that is when a small amount of data is inserted into a file or deleted from a file, file chunking yields totally different result after insertion. This simply means if any of the data of the chunked files being altered during the chunking process, the final result will not be the same file as before the file was even being chopped down into pieces of data.

Meanwhile, variable size data was introduced to tackle the problem of fixed size chunking process. In variable size chunking, chunking boundary is determined based upon the content of a chunk not upon the offset from the beginning of a file. Therefore, after inserting new data into the file, most of the chunks remain unaffected.

File chunking also being used in peer-to-peer file sharing system at the client side of the architecture. Example of peer-to-peer file sharing system is torrent. File chunking is already used today's system to improve response time for downloading large files (Dumitriu, Knightly, Kuzmanovic, Stoica, & Zwaenepoel, 2005).

## **2.6 Binding application to a specific interface**

Binding IP address to a network interface has long been created. In Windows IP binding occurred by software injecting itself into another application and alter how certain Windows Sockets calls are made, allowing user to force the other application to use a specific network interface or IP address. IP binding has been very useful in an environment where there are multiple interfaces and application inside the computer has no specific option for binding to a specific interface.

## **2.7 Transmission Control Protocol and User Datagram Protocol**

Transmission Control Protocol (TCP) is known as the reliable connection because it will have the three-way-handshake before a connection between a target and destination could happen. TCP connection often being used in a very sensitive information transfer and the disadvantage of using TCP is that it is time consuming. User Datagram Protocol (UDP) on the other hand is a non-reliable connection because it does not care whether the packet sent to the destination is received or not at all. UDP often use in an environment where less sensitive and time constraint. UDP does not establish three-way-handshake and that is why it is faster than TCP.

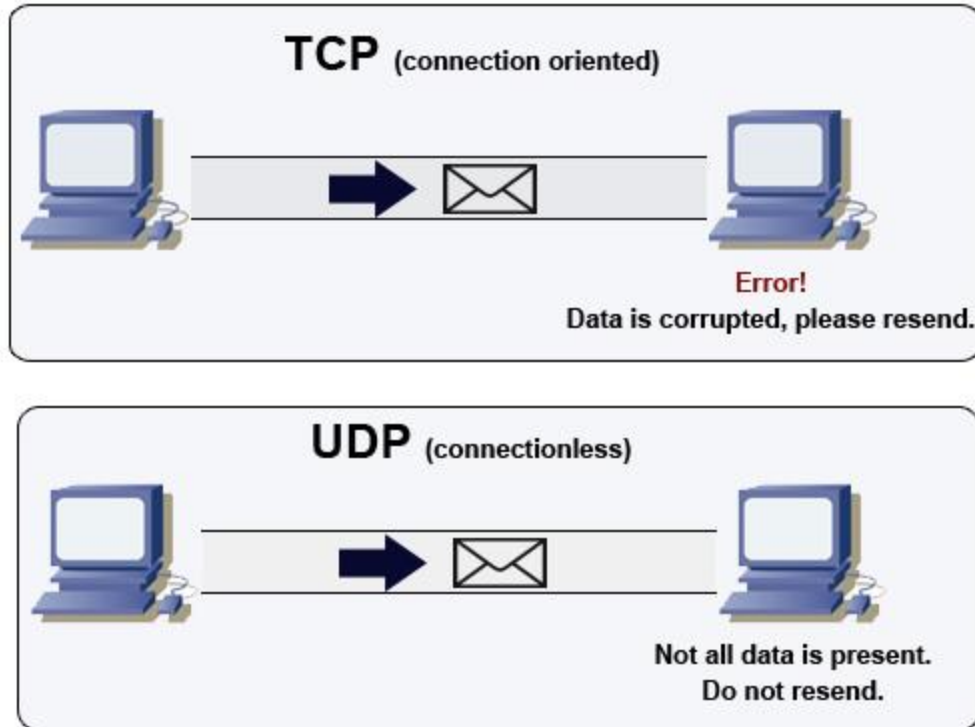


Figure 10: Difference between TCP and UDP

## 2.8 Download simultaneously using multiple internet connection theory

Download file size =  $x$  Byte

Average Download Speed =  $y$  Byte/second

Download Using Single Internet Connection:

Estimated download time,  $t = \frac{x \text{ Byte}}{y \text{ Byte/sec}}$

$$t = \frac{x}{y} \text{ sec}$$

Download using  $n$  Internet Connection simultaneously:

Estimated download time,  $t_{\Sigma n} = \max\left(\frac{\frac{x \text{ Byte}}{n}}{y_1 \text{ Byte/sec}}, \frac{\frac{x \text{ Byte}}{n}}{y_2 \text{ Byte/sec}}, \dots, \frac{\frac{x \text{ Byte}}{n}}{y_n \text{ Byte/sec}}\right)$

$$\text{Percentage of increase Download speed \%} = \frac{t_n - t_{\Sigma n}}{t_n} \times 100$$

Consider that using first internet connection with average download speed is 40kB/s and the download file size is 5MB, the estimated download time will be 125 seconds. Second internet connection has average download speed 50kB/s and download same file which have size of 5MB that will produce estimate download time is 100 seconds. It obviously show that the second internet connection is much faster than first internet connection, but what if we have both internet connection and allowable to use both of them simultaneously? Based on the file size of 5MB and having two internet connections each have average download speed of 40kb/s and 50kb/s. Since we use two internet connections, the first internet connection will download the first half of download file and at the same time, second connection will download the other half of the file which divides both of them to be 2.5MB. By downloading each of the files concurrently, the first internet connection will have estimate download time of 63 seconds and second internet connection have estimate download time of 50 seconds. By taking the longest download time as new estimate time download which is 63 seconds. Then, calculate back the download speed rate using two internet connections that equal of 79kB/s that is much higher than actual speed of both connections.

## Chapter 3: Methodology

### 3.1 Research Methodology

Before going into designing software, it is important to know what type of approach that will be used. Agile methodology is the most appropriate approach that is going to be used in order to cope with iterative and incremental requirement of the software.



Figure 11: Cycle of Project Phases

This research methodology will focus on articles, journals, reviews of books, internet resource and other resources that are useful in developing this project.

### 3.1.1 Planning

For planning stage, the author is putting together all the problem statements on developing the project and thinks on the way this project will address the problems. The author also planning to do reality analysis in order to make sure that the project meets the requirement that is going to solve the problems.

Data gathering process and research is going to be executed in order to gather valuable information with respect to the project title and scope. The planning activities from this stage are used as a baseline to monitor the project progress by stages. In order for the author to stay on schedule, a Gantt chart has been developed to make sure task is achieved within allocated time.

Planning phase is the most important part where all of the basic understanding will be built. To dive deeper into the topic, the author has consulted network analysis to ask information on the project title and methodology. Based on the result of consultation between the author and network analysis, the author is able to clarify the requirements of the project that needs to be done.

From the research gathered, author has identified main functions that needed in the system to make the project success. The following main functions are:

- HTTP URL Connection
  - To read data from URL by making request from HTTP Server
- Proxy Server
  - Allows client to request data from any server by using middle person to get the resource requested.
- File Chunking and Merging

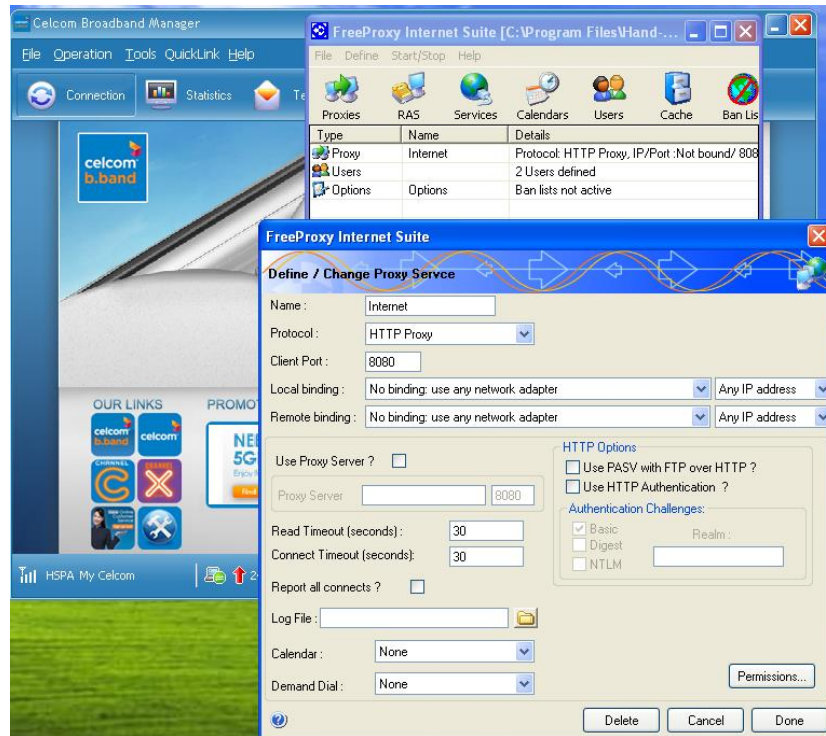
- Download requested file by parts based on user connection. It will request data start at specified byte and end at specified byte and save as temporary file part in the computer. Merging is a process where all the file part will be combining together to be completed download file.
- Threading
  - To make the download execution becomes different threads and allowing download running at a same time.

### **3.1.2 Analysis**

The purpose of data analysis is to find the solution for all the problem statement of the project. Research is the most crucial in this stage in order to analyze the requirement needed for project. Journals, web-site articles, newspapers and books are widely used to find out the solution and strengthen the knowledge about the subject of research. All the related works has been gathered and reviewed in literature review part which aimed to get the importance and benefits of this project.

The information gathered during planning stage is being analysed to cope with the problem statement and the objective of the project. During this stage, the author should identify the steps that need to be done after clear justification of problem statement and objectives. Example of steps that can be done is setting up proxy server in virtual machine in order to test the internet connection with routed networks.





**Figure 12: Creating proxy for Celcom Broadband in Virtual Machine**

A download manager is a computer program dedicated to the task of downloading possibly unrelated stand-alone files from Internet into user's computer. To make a download manager, HTTP Connection will be used to request data from the internet. As for browser the data will represent in HTML and being decoded by the browser to show graphical view of the code to user while mostly of other files such as application(.exe), video(.avi,.mp4) will be treated as unrecognized file and send directly to user's computer. Data connection from HTTP server will receive bytes by bytes depends on the internet connectivity of the user which the faster the internet connection the faster transfer rate of downloaded file. Requested HTTP connection basically will return the HTTP header first then the content of the file. Example of HTTP request of html file:

```
HTTP/1.1 200 OK
Date: Mon, 23 May 2005 22:38:34 GMT
Server: Apache/1.3.3.7 (Unix) (Red-Hat/Linux)
Last-Modified: Wed, 08 Jan 2003 23:11:55 GMT
Etag: "3f80f-1b6-3e1cb03b"
Content-Type: text/html; charset=UTF-8
Content-Length: 131
Connection: close

<html>
<head>
  <title>An Example Page</title>
</head>
<body>
  Hello World, this is a very simple HTML document.
</body>
</html>
```

**Figure 13: Response from HTTP Server**

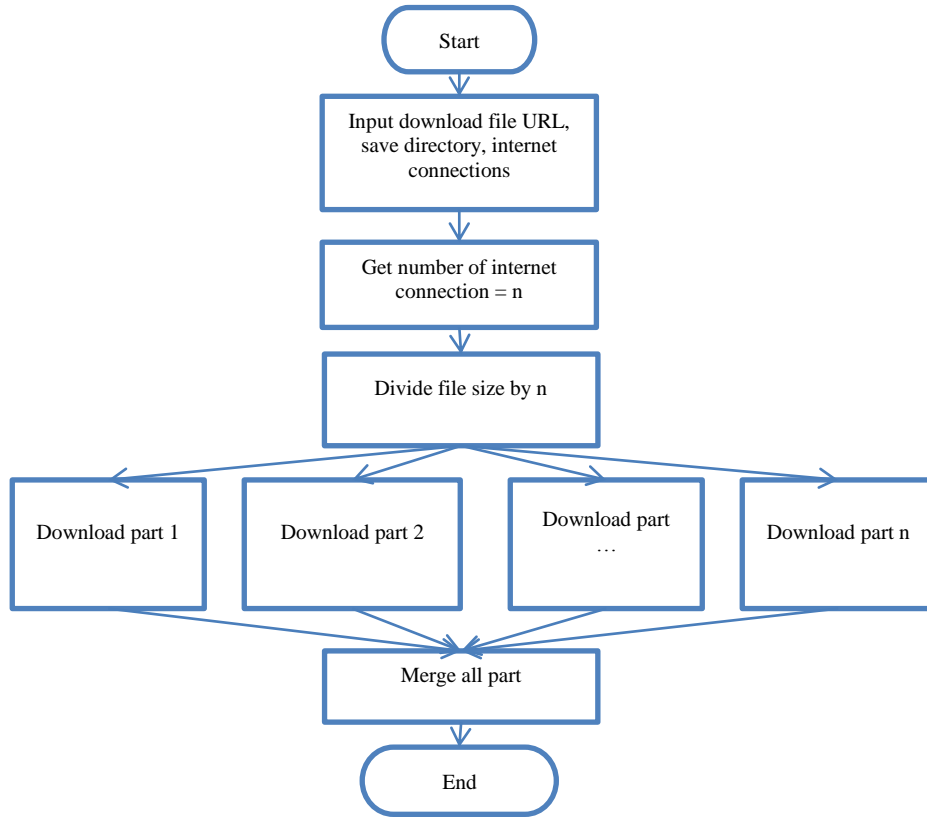
From the figure 7, the HTTP server will give responses for the URL requested started with HTTP header that give some information about the requested file. In this figure, it shows that the connection give HTTP code of 200 which tell the request for the following requested file existing in the server. It also gives information on file type and length (size) of the file.

Since author used JAVA platform to make download request to the HTTP Server, he used Class `HttpURLConnection` which is an object subclass from `java.net.URLConnection`. This class also provides method to get the requested data at a specified start and end bytes.

### 3.1.3 Design

This is the phase where the real system or software is being developed. The author has done some research on how to integrate multiple internet connection. The tools such as Netbeans will be used to develop the user interface of the software. Along the development of user interface, there will always be an important customization that needs to be done in order to make sure the system is still in scope and objective. Realizing the importance of user interface, the author focuses on to capture interest of end user by making the software appeal to them. It is crucial to keep the user interface as simple as possible due to the fact that end user does not want to know underlying architecture and only important function needs to be shown to the user.

To make the development phase much easier, the flowchart for the system is needed. By breaking down system needs based on the earlier part, this system flowchart as follow:



**Figure 14: Flow chart of the system**

List of functions also have been identified based on the flow chart. The lists of functions are function to get the download file information, download file, download by thread and merging and cleaning function and one optional function which a statistic function.

First function is a function that will validate the download URL input as it available or not in the internet. If the validation is success then the function will continue to get information about download link and return data of file name, file size and file extension as parameter for the next function while if validation is fail then it will halt the system.

Next function is to download the file from the validated download URL that will download chunk of data file based on several parameters. The parameters needed are starting and end byte location, proxy information functioning as internet connection, save directory. During this execution of this function, return value of speed rate will be the output until function finish download requested file and at the end of execution, the finish state will be send to main functions.

Third function that needed for this system is a merging function that has some parameters that will return a complete file. The parameters are number of part and the part file name. Basically this function will take the first part and merge it with second part and first and second part will merge with third part and continuously merging until all part has merge. This function also will return the finish state to main function.



**Figure 15: Using loop to merge the chunked file**

By using the functions above and based on parameter needed, another function which is the main function needed to call all of the previous functions. This main function will used all the return value from each of these function and integrate it with the UI.

Understanding on the functions itself have given overview on the structure needed on the system, below is the structure for this system:

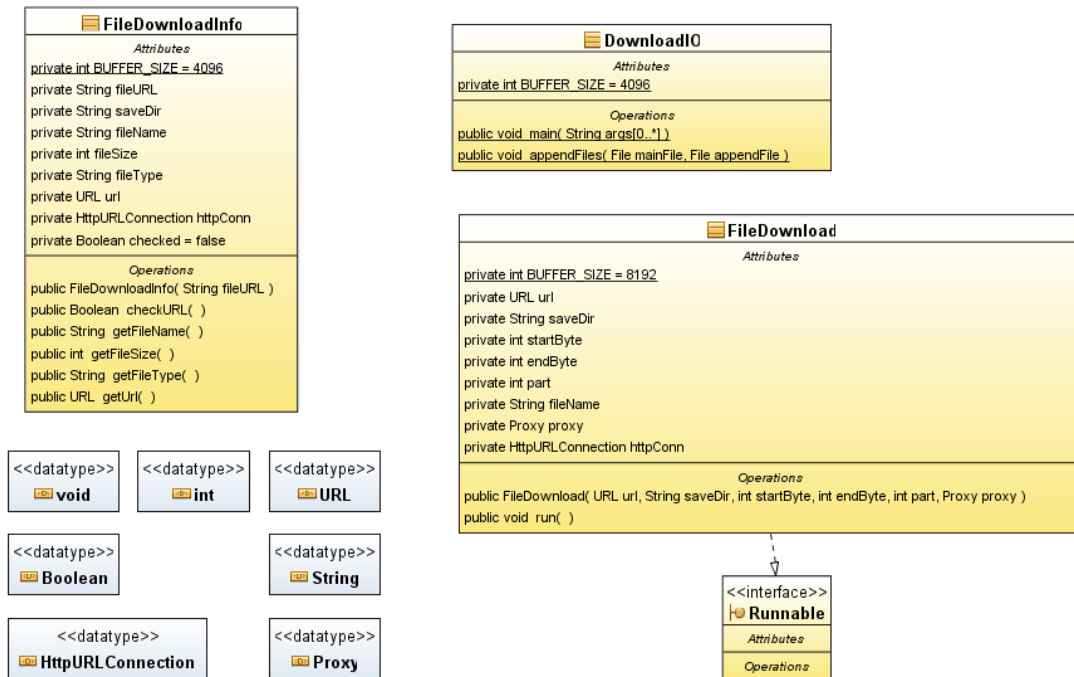


Figure 16: Class Diagram for download manager

### 3.1.4 Development

This stage is important to see the results of initial development of the software. There will also tools that are going to help the author to monitor the result from the developed software. The system will be finalized after the objective has been achieved which is increased download speed by using multiple internet connection and load between the connection is balanced. Based on the specification given on the design phase the function created as follow:

#### Download Function

FileDownloadInfo.java

```

import java.io.IOException;
import java.net.HttpURLConnection;
import java.net.URL;

public class FileDownloadInfo {
    private static final int BUFFER_SIZE = 4096;
    private String fileURL;
    private String saveDir;
    private String fileName;
    private int fileSize;
    private String fileType;

    private URL url;
    private HttpURLConnection httpConn;

    private Boolean checked = false;

    public Boolean isFileDownloadExist(){
        return checked;
    }

    public FileDownloadInfo(String fileURL) {
        this.fileURL = fileURL;
        try {
            checked = checkURL();
        } catch ( IOException e){

        }
    }

    public Boolean checkURL() throws IOException{
        url = new URL(fileURL);
        httpConn = (HttpURLConnection) url.openConnection();
        int responseCode = httpConn.getResponseCode();

        if (responseCode == HttpURLConnection.HTTP_OK) {
            String disposition = httpConn.getHeaderField("Content-Disposition");
            fileType = httpConn.getContentType();
            fileSize = httpConn.getContentLength();

            if (disposition != null) {
                // extracts file name from header field
                int index = disposition.indexOf("filename=");
                if (index > 0) {
                    fileName = disposition.substring(index + 10,
                        disposition.length() - 1);
                }
            } else {
                // extracts file name from URL
                fileName = fileURL.substring(fileURL.lastIndexOf("/") + 1,
                    fileURL.length());
            }

            httpConn.disconnect();
            return true;
        } else {
            System.out.println("Server replied " + responseCode + " Error");
            return false;
        }
    }

    public String getFileName() {
        return fileName;
    }

    public int getFileSize() {

```

```

        return fileSize;
    }

    public String getFileType() {
        return fileType;
    }

    public URL getUrl() {
        return url;
    }
}

```

This code used to check response from HTTP Server to know the file exists and to save certain information about the file such as filename and file size.

### FileDownload.java

```

import java.io.File;
import java.io.FileOutputStream;
import java.io.IOException;
import java.io.InputStream;
import java.net.*;
import java.util.Observable;

public class FileDownload extends Observable implements Runnable {
    private static final int BUFFER_SIZE = 8192;

    private URL url;
    private String saveDir;
    private int startByte;
    private int endByte;
    private int part;
    private String fileName;
    private Proxy proxy;
    private long totalBytes;
    private double progressPercentage;
    private long speedRate;

    private HttpURLConnection httpConn;

    public long getSpeedRate(){
        return speedRate;
    }

    public long getTotalBytes(){
        return totalBytes;
    }

    public double getProgressPercentage(){
        return progressPercentage;
    }

    public int getPart(){
        return part;
    }

    public FileDownload(URL url, String saveDir, int startByte, int endByte, int part, Proxy proxy) {
        this.url = url;
    }
}

```



```

this.saveDir = saveDir;
this.startByte = startByte;
this.endByte = endByte;
this.part = part;
this.proxy = proxy;

String fileURL = url.toString();
fileName = fileURL.substring(fileURL.lastIndexOf("/") + 1, fileURL.length());

}

@Override
public void run() {
    Timer timer = new Timer();
    Thread t = new Thread(timer);
    t.start();
    try {
        httpConn = (URLConnection) url.openConnection(proxy);

        String byteRange = startByte + "-" + endByte;
        httpConn.setRequestProperty("Range", "bytes=" + byteRange);
        InputStream inputStream = httpConn.getInputStream();
        String saveFilePath = saveDir + File.separator + fileName + ".part" + part;
        System.out.println(saveFilePath);

        // opens an output stream to save into file
        FileOutputStream outputStream = new FileOutputStream(saveFilePath);

        int bytesRead = -1;
        byte[] buffer = new byte[BUFFER_SIZE];
        totalBytes = 0;
        int size = endByte - startByte;
        while ((bytesRead = inputStream.read(buffer)) != -1) {
            outputStream.write(buffer, 0, bytesRead);
            totalBytes += bytesRead;
            //DecimalFormat twoDForm = new DecimalFormat("#.00");
            progressPercentage = ((double)totalBytes)/((double) size) * 100;
            //System.out.println("Part " + part + " : " +twoDForm.format(progressPercentage) +
"%%");
            setChanged();
            notifyObservers();
        }
        outputStream.close();
        inputStream.close();
        //timer.stop();
        //System.out.println("File part " + part + " downloaded");
        httpConn.disconnect();
    } catch( IOException e ){

    }
    timer.stopExecuting();
}

private class Timer implements Runnable {
    private long prevBytes = 0;
    private volatile boolean execute;
    @Override
    public void run(){
        this.execute = true;
        while(this.execute){
            try {
                speedRate = totalBytes - prevBytes;
                setChanged();
                notifyObservers();
                prevBytes = totalBytes;
                Thread.sleep(1000);
            }

```

```

        } catch (InterruptedException ex) {
            this.execute = false;
        }
    }
}
public void stopExecuting() {
    this.execute = false;
}
}
}
}

```

This code is the main function which is to download the data from specified byte and download it concurrently.

```

import java.io.File;
import java.io.FileInputStream;
import java.io.FileOutputStream;
import java.io.IOException;
import java.net.InetSocketAddress;
import java.net.Proxy;
import java.util.logging.Level;
import java.util.logging.Logger;

public class DownloadIO {
    private static final int BUFFER_SIZE = 4096;

    public static void main(String[] args) {
        String fileURL = "http://jaist.dl.sourceforge.net/project/winmerge/stable/2.14.0/WinMerge-2.14.0-Setup.exe";
        String saveDir = "D:\\Test";
        FileDownloadInfo dl = new FileDownloadInfo(fileURL);
        System.out.println(dl.getFileName());
        System.out.println(dl.getFileType());
        System.out.println(dl.getFileSize());
        System.out.println(dl.getUrl().toString());

        Proxy[] proxy = {Proxy.NO_PROXY, new Proxy(Proxy.Type.HTTP, new
        InetSocketAddress("164.0.2.124", 8080))};
        int fileSize = dl.getFileSize();
        int maxPart = proxy.length;
        int[][] parts = new int[maxPart][3];
        FileDownload[] dls = new FileDownload[maxPart];
        Thread[] t = new Thread[maxPart];
        for ( int i = 0 ; i < maxPart; i++){
            if ( i == 0 ){
                parts[i][1] = 0;
                parts[i][2] = fileSize / maxPart;
            }
            else if ( i == maxPart - 1){
                parts[i][1] = parts[i-1][2];
                parts[i][2] = fileSize;
            }
            else {
                parts[i][1] = parts[i-1][2];
                parts[i][2] = parts[i][1] + (fileSize / maxPart);
            }
            parts[i][0] = i + 1;
            t[i] = new Thread(new
            FileDownload(dl.getUrl(), saveDir, parts[i][1], parts[i][2], parts[i][0], proxy[i]));
            t[i].start();
        }
    }
}

```

```

}
for ( int i = 0 ; i < maxPart; i++){
    try {
        t[i].join();
    } catch (InterruptedException ex) {
        Logger.getLogger(DownloadIO.class.getName()).log(Level.SEVERE, null, ex);
    }
}
String partPath = saveDir + "\\\" + dl.getFileName();
String mainSaved = partPath + ".part1";
File mainFile = new File(mainSaved);
File appendFile;
for ( int i = 1 ; i < maxPart; i++){
    appendFile = new File(partPath + ".part" + (i+1));
    System.out.println("Merging file part" + (i+1));
    appendFiles(mainFile,appendFile);
    appendFile.delete();
}
mainFile.renameTo(new File(partPath));
System.out.println("Download complete");
}
public static void appendFiles(File mainFile, File appendFile){
    if (!mainFile.exists() || !appendFile.exists()) {
        System.out.println("Failed");
        return;
    }
    try {
        byte[] readData = new byte[4096];
        FileInputStream fis = new FileInputStream(appendFile);
        FileOutputStream fos = new FileOutputStream(mainFile, true);
        int i = fis.read(readData);
        fos.write(readData, 1, i-1);
        i = fis.read(readData);
        while (i != -1) {
            fos.write(readData, 0, i);
            i = fis.read(readData);
        }
        fis.close();
        fos.close();
    } catch (IOException e) {
        System.out.println(e);
    }
}
}
}

```

Simple interface that used both of FileDownloadInfo.java and FileDownload.java, it will ask input from user for the download link.



Figure 17: Simple interface

## Graphical User Interface

By using Netbeans tool to create GUI the interface:

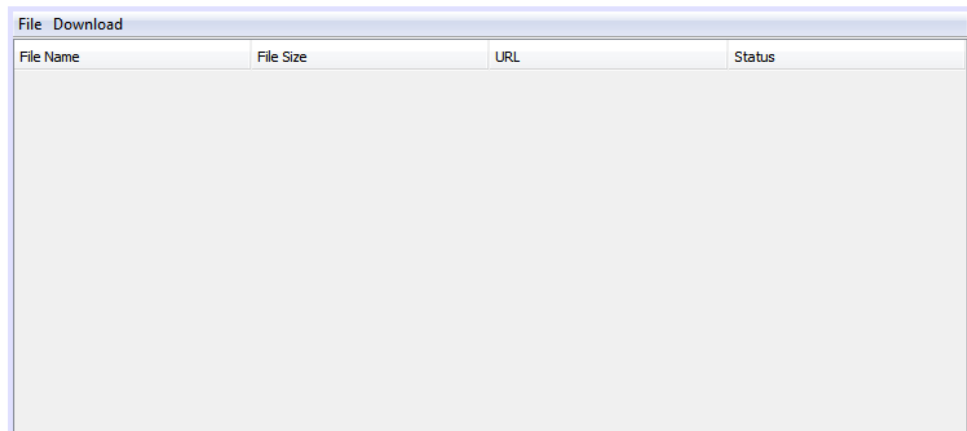


Figure 18: Main Download Interface

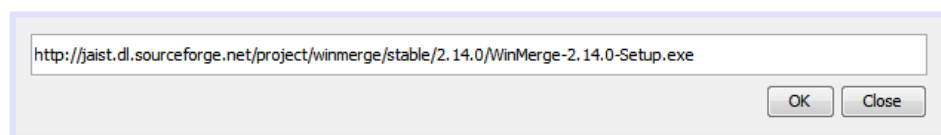


Figure 19: URL input

Figure 20: Download Information

Figure 21: Download progress

## Integration function and GUI

### Sending URL input into Download Information Interface

```
final DownloadInformationGUI dlInfoForm = new DownloadInformationGUI(urlField.getText());
java.awt.EventQueue.invokeLater(new Runnable() {
    public void run() {
        dlInfoForm.setVisible(true);
    }
});
```

Confirmation of download and getting input on internet connections and save directory

```
String saveDir = saveDirField.getText();
String[] proxyList = internetsField.getText().split(",");
Proxy[] proxy = new Proxy[proxyList.length];
for ( int i = 0 ; i < proxyList.length; i++ ){
    if ( proxyList[i].equals("local")){
        proxy[i] = Proxy.NO_PROXY;
    } else {
        String temp2[] = proxyList[i].split(":");
        proxy[i] = new Proxy(Proxy.Type.HTTP,new InetSocketAddress(temp2[0],
Integer.parseInt(temp2[1])));
    }
}
DownloadProgress dp = new DownloadProgress(url,fileName,fileSize,proxy,saveDir);
new Thread(dp).start();
```

Download by part by using java thread

```
this.setVisible(true);
int maxPart = proxy.length;
fileNameLabel.setText(fileName);
//Parts Splitter with data of [0] = Partnumber , [1] = Start offset, [2] = End Offset
int[][] parts = new int[maxPart][3];
//FileDownload[] dls = new FileDownload[maxPart];

//Create array of thread based on connection
Thread[] t = new Thread[maxPart];
long startTime = System.currentTimeMillis();
//Splitting download into multiple part
for ( int i = 0 ; i < maxPart; i++){
    if ( i == 0 ){
        parts[i][1] = 0;
        parts[i][2] = fileSize / maxPart;
    }
    else if ( i == maxPart - 1){
        parts[i][1] = parts[i-1][2];
        parts[i][2] = fileSize;
    }
    else {
        parts[i][1] = parts[i-1][2];
        parts[i][2] = parts[i][1] + (fileSize / maxPart);
    }
    parts[i][0] = i + 1;
    System.out.println(parts[i][1]+"|"+parts[i][2]);
    FileDownload startDownload = new
FileDownload(url,saveDir,parts[i][1],parts[i][2],parts[i][0],proxy[i]);
    startDownload.addObserver(this);
    t[i] = new Thread(startDownload);
    t[i].start();
}

//Joining each file part synchronously
for ( int i = 0 ; i < maxPart; i++){
    try {
```

```

        t[i].join();
    } catch (InterruptedException ex) {
        System.err.println(ex);
    }
}
String partPath = saveDir + "\\\" + fileName;
String mainSaved = partPath + ".part1";
File mainFile = new File(mainSaved);
File appendFile;
for ( int i = 1 ; i < maxPart; i++){
    appendFile = new File(partPath + ".part" + (i+1));
    System.out.println("Merging file part" + (i+1));
    appendFiles(mainFile,appendFile);
    appendFile.delete();
}
mainFile.renameTo(new File(partPath));

long currentTime = System.currentTimeMillis();
//System.out.println(startTime + "|" + currentTime);

int timeTaken = (int) ((currentTime-startTime)/1000);
System.out.println(fileSize);
int avgSpeed = fileSize/timeTaken;
JOptionPane.showMessageDialog(this,"Time Taken to complete download: " +
secondsToString(timeTaken));

```

### 3.1.5 Implementation

To make sure the system can achieved the objective, this phase is needed to complete a system follow the specification given on the design phase and being built during development phase. This phase also is to find bugs on the system that can affect the system functionality and user experience on the system. Below is one of the testing that has been done in this project.

#### Download using two internet connections

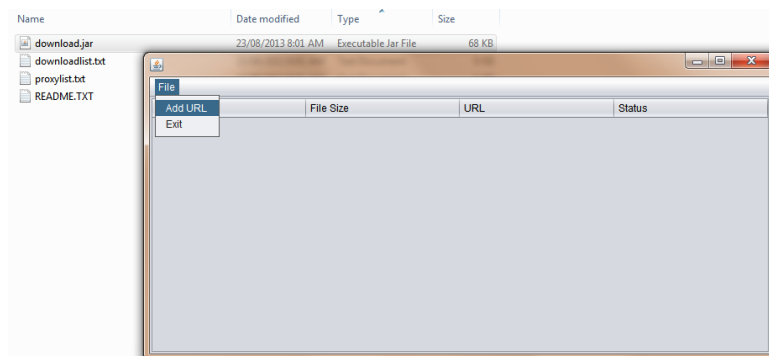


Figure 22: The executable program for this project

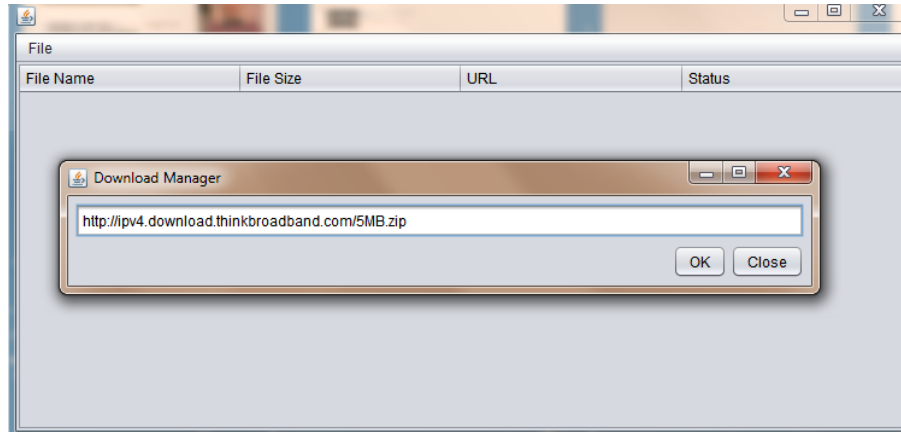


Figure 23: Download sample file from internet

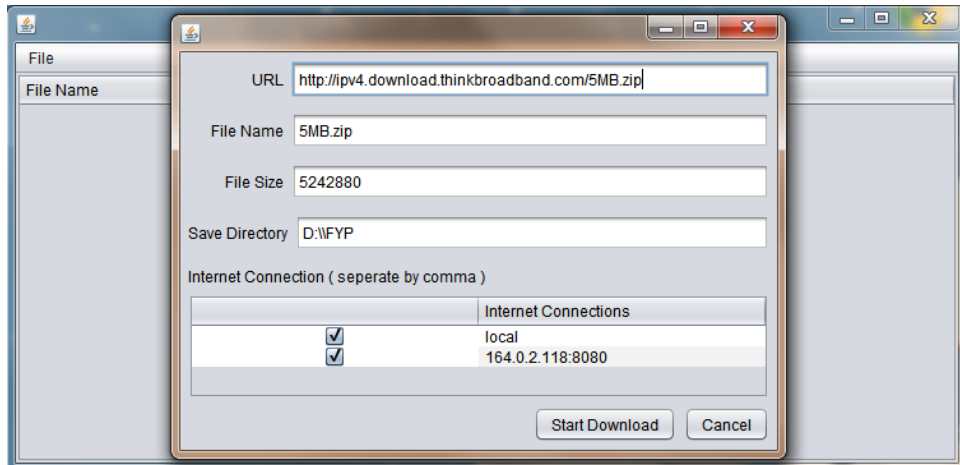


Figure 24: Download Information

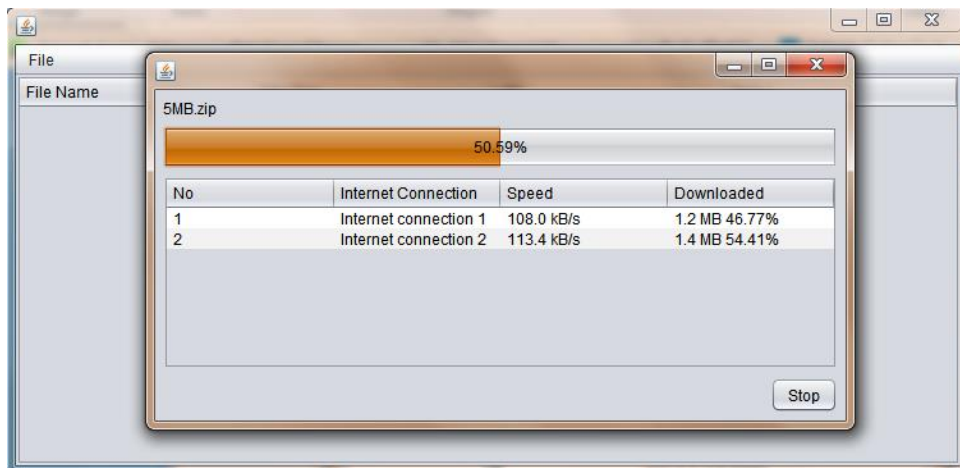


Figure 25: Download Progress



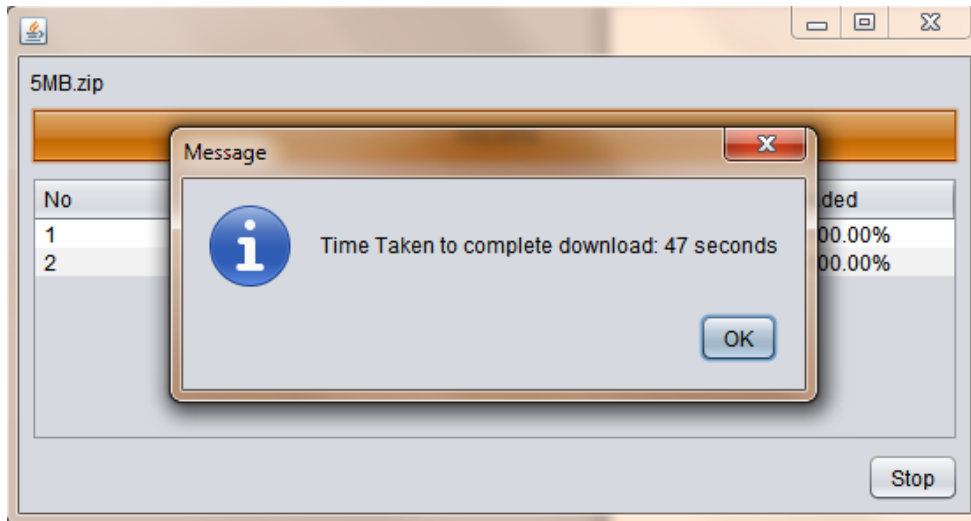


Figure 26: Successfully download file

### Download using Firefox

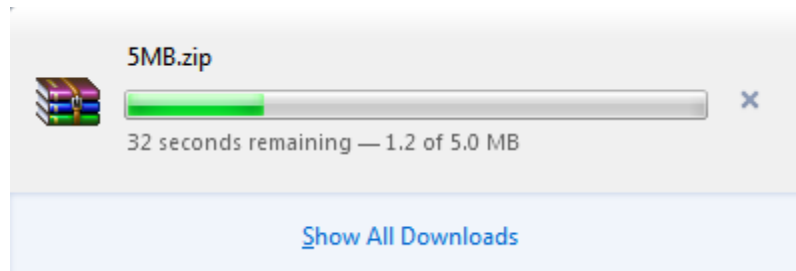


Figure 27: Download file using fire

Time taken for this download is taken manually by using stopwatch and to download this file using Firefox for 62 seconds.

### Compare downloaded files

Basically the downloaded file using Firefox is the reference file that compare with downloaded file using this project system. This test is to compare whether the file downloaded give the same output or not.

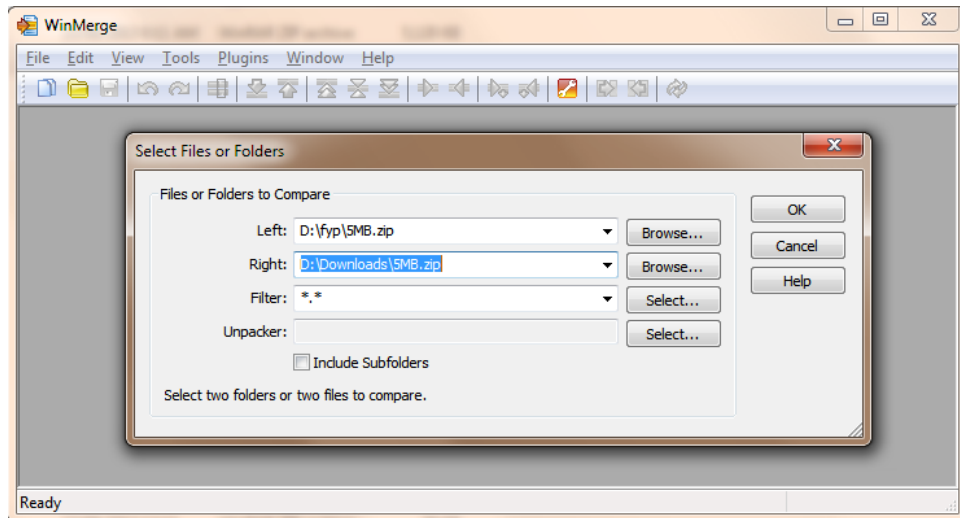


Figure 28: Left side - file download using this project downloader, right side - file download using Firefox

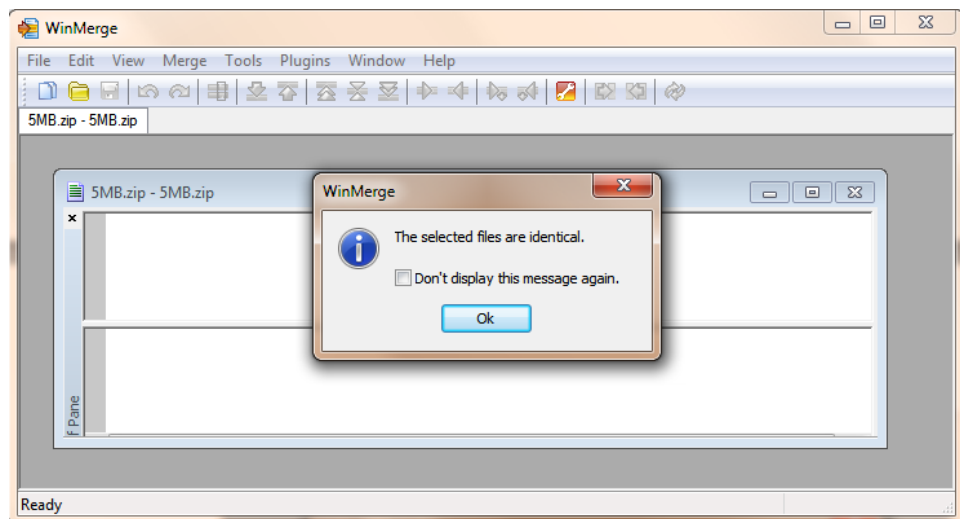


Figure 29: Show the file is identical

This show that the file downloaded is same which show that there is no corruption of data by using this project downloader.

### 3.2 Tool Used

- Hardware
  - Ethernet adapter

An Ethernet adapter will be used to acquire internet connection from the fixed line internet connection that using DSL cable or fiber optics.

- Wireless adapter

External wireless adapter is going to be used in order to simulate multiple internet connection. Each network card either wired or wireless is able to grab one internet connection at a time

- USB Broadband Dongle

The dongle will be used to connect a computer to the internet with respect to Malaysia's Internet Service Provider (ISP) such as Maxis, Celcom and Digi. Each ISP will have their own USB dongle that user has to connect to their computer in order to receive internet connection.

- Software

- Netbeans IDE/Eclipse IDE

This software will be used to create a system interface using Java programming language. Netbeans have provided an easy drag and drop feature in order to create an ergonomic user interface

- JAVA

Java will be used as the programming language in order to build the prototype.

- Virtual Machine

This software will be used for testing purposes where a proxy server will be setup in virtual machine.

- Win Merge

This software will be used for testing purposes to compare every byte from the original download file and file downloaded from prototype.

- MD5 File Comparison Tool

Compare files using MD5 Hash

## Chapter 4: Result and Discussion

There are some challenges that need addressed in order to reduce the negative impact on the performance of the prototype. Here are the findings based on the observation during project testing:

### 4.1 Several factors need to take measure before making decision

**Throughput:** measurement of how much data can be moved through the connection in a certain amount of time. In order to ensure optimum usage of all internet connection a user has, the prototype need to measure the throughput of the connection.

**Latency:** measurement of how long it takes for and internet traffic to travel between two points. Higher latency means the connection is less reliable and it takes more time to travel between points.

**Recovery method when one internet connection is disconnected**

During the download of the files using multiple internet connection, there is a possibility that any of those connection suddenly disconnected. This would somehow interrupt the download process because the file has already been chunked and ready to be downloaded by respective internet connection. Recovery method to overcome this problem should be taken into consideration because the data might be lost if the chunked file is not being downloaded using disconnected internet connection.

The recovery method somehow has to address problem of disconnected one of the internet connection by redirect the packet to be sent to another internet connection. For example, if a broadband dongle suddenly does not work, the packet is able to be redirect to another working internet connection so that the process of downloading a file will run smoothly.

## 4.2 Data result

Result of download file of 1MB using 3 internet connections

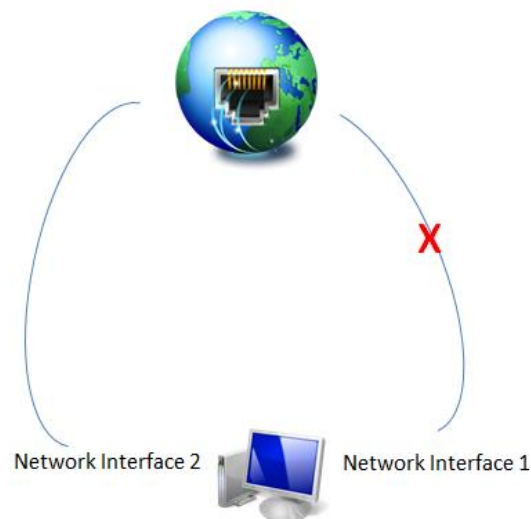
Internet Connection	Average Download Speed ( KB/s )	Downloading Time ( s )	Percentage of Difference between single and Multiple connection
<b>UTP LAN 1</b>	7.06	142	78%
<b>UTP LAN 2</b>	27.83	36	16%
<b>Celcom Broadband</b>	17.27	58	48%
<b>UTP LAN 1 &amp; UTP LAN 2 &amp; Celcom Broadband</b>	33.34	30	0%

Video demo of this system function as the link below:

<http://bit.ly/17WXqxA>

#### 4.3 Multiple default gateway can cause connectivity problems

When multiple default gateways are used in TCP/IP configuration options on a Windows NT computer, connectivity to computers on remote networks may be lost.



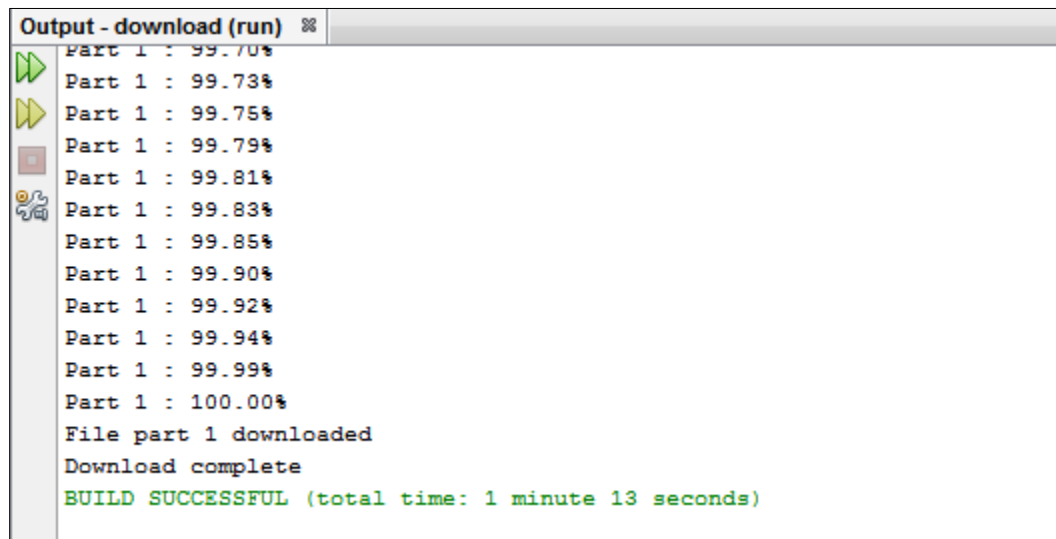
**Figure 30: Network interface 1 detected as dead by TCP**

By referring to the above image, the computer is using two different network interfaces. When using more than one interface, this means that the computer has more than one gateway. This would probably cause connectivity problem when the gateway is detected as dead by TCP, it will lost all of its connection and session created before. Even though the second network interface is present, it is

not aware of the other routes managed by the first gateway if routers do not exchange routing information to each other. In order to prevent this problem from happening, the prototype should be able to add route into the computer so that the second network interface can learn about all the routes in first network card.

#### 4.4 Reduce downloading times

Based on testing by downloading file from this URL <http://jaist.dl.sourceforge.net/project/winmerge/stable/2.14.0/WinMerge-2.14.0-Setup.exe> that size 6.3MB that require 73 sec to download by using one internet connection that have average speed of 83KB/s while when using 2 internet connection the download only need 43 sec which combined both of internet connection speed and have average of 146KB/s. It also saves half of bandwidth used for my Celcom Internet Connection for 3MB.



```
Output - download (run) %
Part 1 : 99.70%
Part 1 : 99.73%
Part 1 : 99.75%
Part 1 : 99.79%
Part 1 : 99.81%
Part 1 : 99.83%
Part 1 : 99.85%
Part 1 : 99.90%
Part 1 : 99.92%
Part 1 : 99.94%
Part 1 : 99.99%
Part 1 : 100.00%
File part 1 downloaded
Download complete
BUILD SUCCESSFUL (total time: 1 minute 13 seconds)
```

Figure 31: Download using 1 Internet Connection - Celcom Broadband

```

Output - download (run)
Part 1 : 97.92%
Part 1 : 97.95%
Part 1 : 98.46%
Part 1 : 98.50%
Part 1 : 99.01%
Part 1 : 99.04%
Part 1 : 99.55%
Part 1 : 99.59%
Part 1 : 100.00%
File part 1 downloaded
Merging file part2
Merging file part3
Merging file part4
Download complete
BUILD SUCCESSFUL (total time: 43 seconds)

```

Figure 32: Download using 2 Internet Connection - Celcom Broadband and UTP LAN

#### 4.5 Downloading time become slower

Based on the several testing that have been made, this showed that the system can give faster download time but when the unconditional thing happen such as second internet connection become slower it will affect the downloading time.

There is some limitation using current theory which is if got two internet connections that have balanced speed rate, it will increase download speed 2 times faster but when only one that have highest speed rate and the second one got lower speed rate, the download time using one internet connection much more faster. So, the speed will only better if total average download time is below than download time using multiple internet connection simultaneously. Below are the equations that show is it better to use multiple or single internet connection.

$$\left( \frac{x \text{ Byte}}{y_1 \text{ Byte/sec}} + \frac{x \text{ Byte}}{y_1 \text{ Byte/sec}} + \dots + \frac{x \text{ Byte}}{y_1 \text{ Byte/sec}} \right) \div n < \max \left( \frac{x \text{ Byte}}{y_1 \text{ Byte/sec}}, \frac{x \text{ Byte}}{y_1 \text{ Byte/sec}}, \dots, \frac{x \text{ Byte}}{y_1 \text{ Byte/sec}} \right)$$

This equation will compare average download time with the optimize download time.



## Chapter 5: Conclusion

This project is focusing on optimizing or taking advantage of many internet connections that a user has in his or her house in order to increase the speed of the internet by identifying the reliability, latency and throughput of each internet connection. It is said that the speed of internet can be increased by combining all the internet connections and appeared to the user as software.

The authors make some observation and analysis on the current downloading system that using single threaded internet connection which usually slow and taken much time to finish downloading a single file. It also comes from the operating system that lacks of system operation that allows the user to use multiple internets at the same time. From the research, the author come up with a theory that can reduce downloading time for a single file download that using multiple internet connection.

To prove the theory, author developed a downloader that allows the user to pick whether to use single or multiple internet connections. Some testing has been made and the downloaded file show an improved download speed rate to download the file. Even though the download time was reduced, but it still needs the internet connection from all connections to have small different gap of the average internet speed to optimize the download speed.

For future improvement, this system can add a pipelining function which allows the finished part to continue download half of the other part that is

currently downloading. This could make the system optimize the download speed much better than current one.

## Chapter 6: Reference

*The Original HTTP as defined in 1991.* (1991). Retrieved 2 19, 2013, from W3C:  
<http://www.w3.org/Protocols/HTTP/AsImplemented.html>

*HTTP/1.1 Specification Errata.* (2004, 10 27). Retrieved 02 2013, 15, from  
[http://skrb.org/ietf/http\\_errata.html](http://skrb.org/ietf/http_errata.html)

*Chunk -- Simple, fast file splitter.* (2006, 12 3). Retrieved 2 15, 2013, from Oldskool PC:  
<http://www.oldskool.org/pc/chunk>

(2008). *White Paper: Download Managers - A Better Downloading Experience.* Conceiva Pty. Ltd.

*Malaysia slips in Internet freedom; same ranking as Libya.* (2012, October 2). Retrieved April 1, 2013, from The Malaysian Insider:  
<http://www.themalaysianinsider.com/malaysia/article/malaysia-slips-in-internet-freedom-same-ranking-as-libya>

A., C., Link, Seattle, Im, H., & Bellevue. (1998). *Patent No. 6,012,096.* United States of America.

Dumitriu, D., Knightly, E., Kuzmanovic, A., Stoica, I., & Zwaenepoel, W. (2005). Denial of Service in Peer-to-Peer File Sharing System. *In ACM SIGMETRICS Performance Evaluation Review*, 38-49.

Gill, P., Jain, N., & Nagappan, N. (2011). Centers, Understanding Network Failures in Data: Measurement, Analysis, and Implications. 12.

Kang, H. (n.d.). *Internet Connection*. University of Texas.

Minges, M., & Gray, V. (2002). *MULTIMEDIA MALAYSIA: Internet Case Study*. International Telecommunication Union (ITU).

Moncada, D. M. (2011, February 23). *Internet Speeds Around The World*. Retrieved April 1, 2013, from Daily Infographic: <http://dailyinfographic.com/internet-speeds-around-the-world-infographic>

Tirumala, A., Cottrell, L., & Dunigan, T. (2003). *Measuring end-to-end bandwidth with Iperf using Web100*. Retrieved April 1, 2013, from Citeseerx: <http://citeseerx.ist.psu.edu/viewdoc/summary?doi=10.1.1.13.1705>

Vicomsoft. (202). *Bandwidth Aggregation, Tteaming and Bonding*.

Youjip Won, R. K. (2008). *PRUN : Eliminating Information Redundancy for Large Scale Data Backup*. 6.

## APPENDICES

