**MID-SEMESTER PROGRESS REPORT**

**ON**

**Implementing Compression Algorithm and Secure Transmission over UDP Protocol**

**Submitted by:**

|  |  |  |
| --- | --- | --- |
| Name | SAP ID | Roll No |
| Harsh Narain Mathur | 500069098 | R100218020 |
| Harshit Chauhan | 500069534 | R100218021 |
| Pranay Mahajan | 500068668 | R100218037 |

**Under the guidance of**

**Mr. Ankit Vishnoi**

Assistant Professor

Department of Systematics

School of Computer Science



**School of Computer Science**

**University of Petroleum and Energy Studies,**

**Dehradun – 248007: Uttarakhand**

**Table of Contents**

[1. Problem Statement 2](#_Toc54282464)

[2. Methodology 2](#_Toc54282465)

[3. Flowchart 3](#_Toc54282466)

[3.1 Use Case Diagram 4](#_Toc54282467)

[4. Algorithm 5](#_Toc54282468)

[5. Implemented Code: 6](#_Toc54282469)

[6. Result 7](#_Toc54282470)

[7. PERT Chart 8](#_Toc54282471)

[8. Scope 9](#_Toc54282472)

[9. Role of each Member: 9](#_Toc54282473)

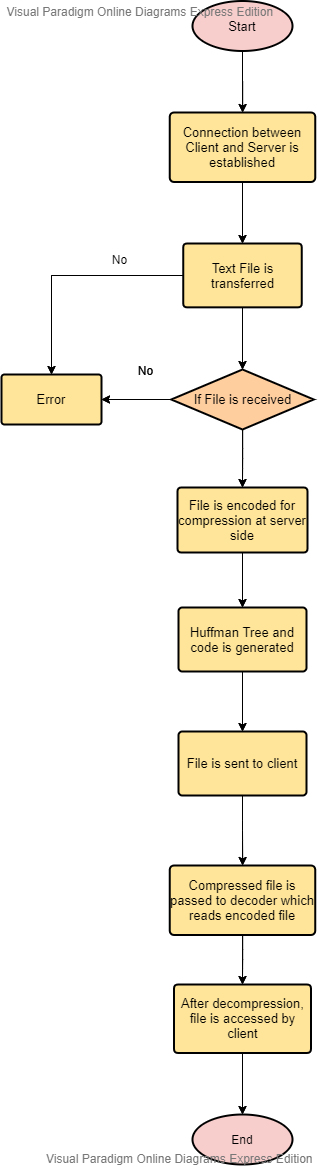
# Problem Statement

This project presents a proposal to make connection between client and server using socket programming and compress the large size of text file into small size without any loss of data.

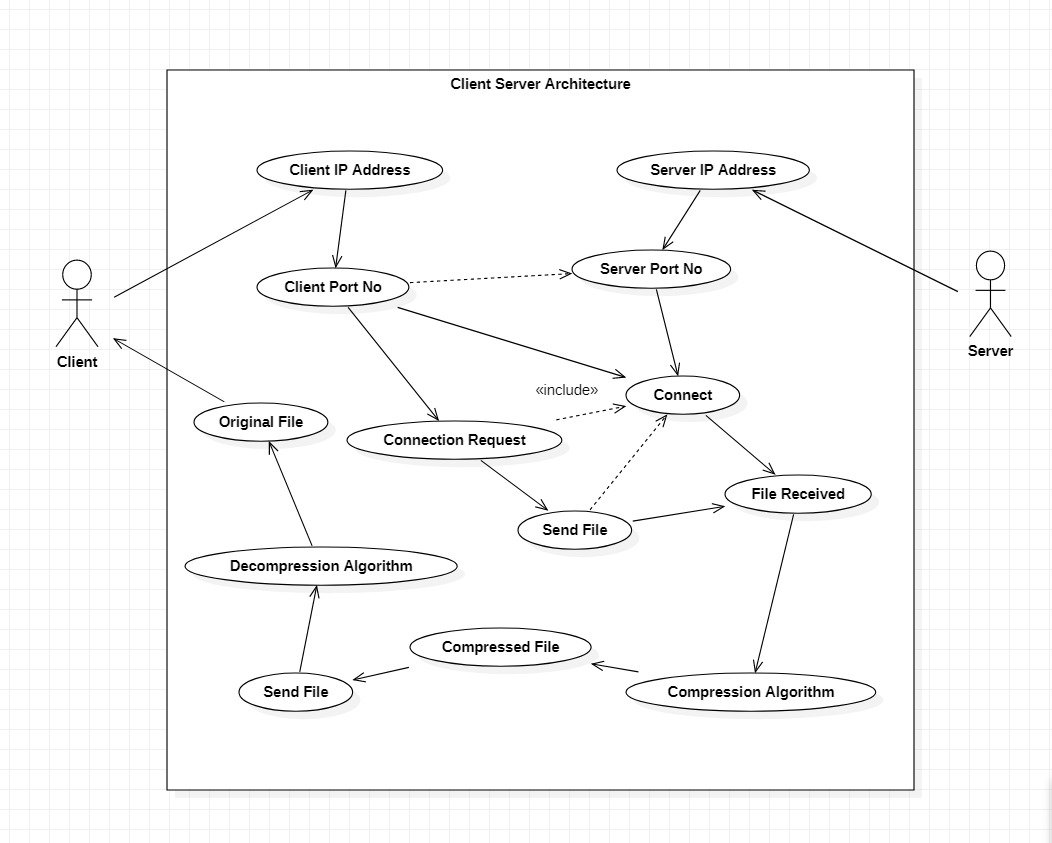
# Methodology

* Firstly, connection between the client and server is established using socket programming. The server opens up a port number to which client will connect for transmission of text file.
* When the connection is established the sample text file is transferred from client to server.
* Server encrypts the received text file and then scans the entire file and stores the contents in the buffer.
* The encrypted file is then transmitted to the client which decrypts the content and is further transmitted to encoder.
* The frequency count of each character is calculated and store it in a table in ascending order.
* To compress the file, prepare the Huffman tree and generate the Huffman code for each character.
* Then the compressed file is passed to the decoder which reads the Huffman encoded file with one binary digit at a time to retrieve the original text.
* After complete decompression, the decompressed file (original contents of file) is generated which can be accessed by the client

# Flowchart



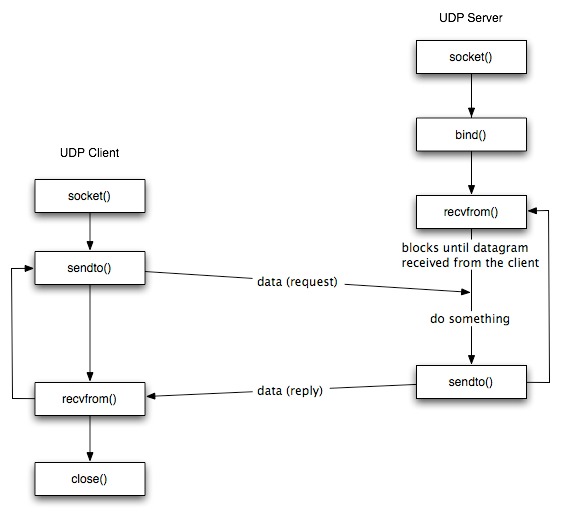
# 3.1 Use Case Diagram



**Fig 1 Use-Case Diagram**

# Algorithm

**Socket Programming**



# Implemented Code:

1. #include<arpa/inet.h> = definitions for internet operations.

The <**arpa/inet.h**> header makes available the type **in\_port\_t** and the type **in\_addr\_t.**

The <**arpa/inet.h**> header makes available the **in\_addr** structure

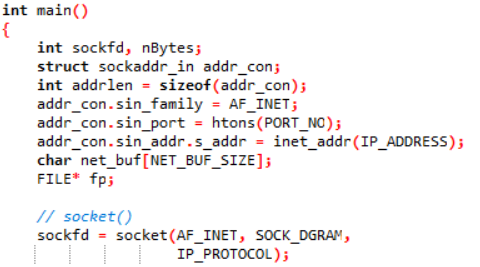
1. #include<netinet/in.h> = Internet address family.

The *<netinet/in.h>* header shall define the **sockaddr\_in** structure that includes at least the following members:

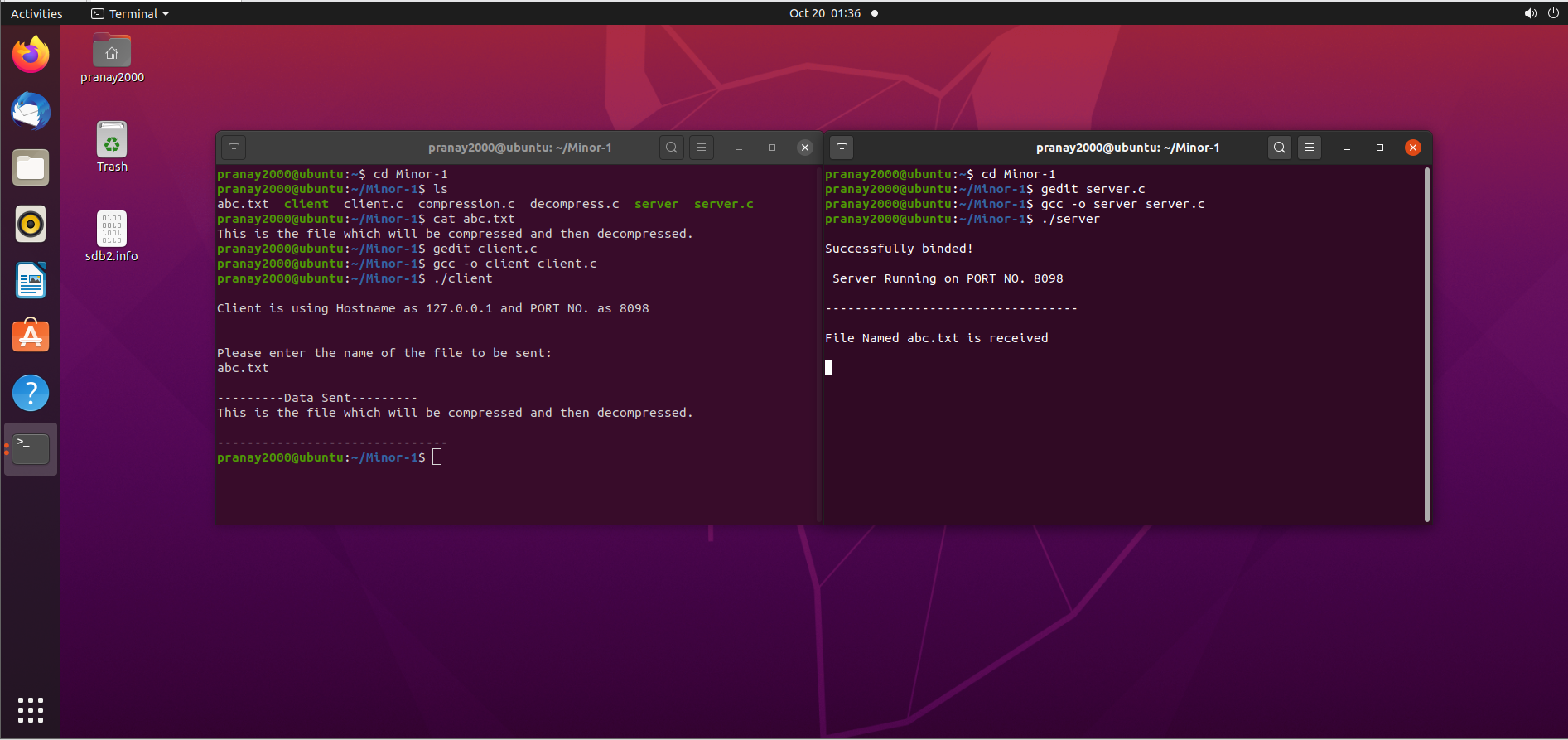
sa\_family\_t sin\_family AF\_INET.

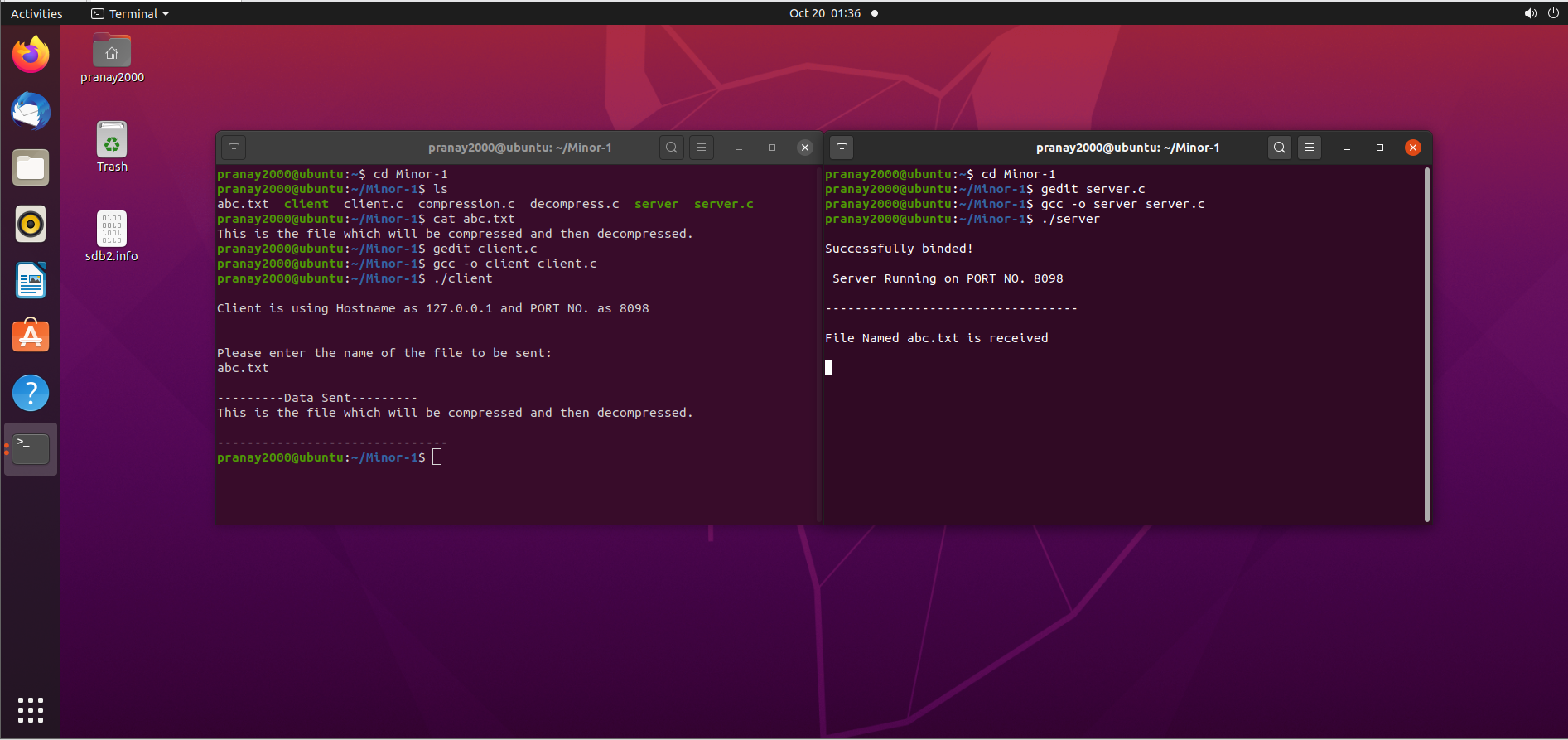
in\_port\_t sin\_port Port number.

struct in\_addr sin\_addr IP address.

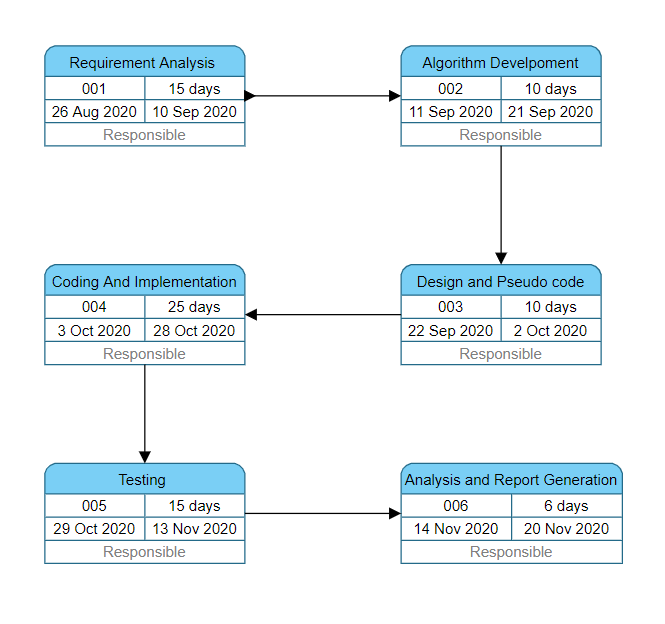
****

# Result





# PERT Chart

****

# Scope

Over the years, there has been a tremendous increase in the amount of digital data produced and transmitted, representing text, images, video, sound, computer programs, etc. With this trend expected to continue, it makes sense to pursue research on compression algorithms.

* Compression algorithms reduce the repetition of data without actual loss of data.
* With its fast-compressed feature, Word Based Compression Algorithm is a great solution to compress text files for better storage, transmission of data that is large and reduce input/output load.
* It helps in reduction of both compression ratio that is defined as the ratio between the uncompressed size and compressed size and transmission time of data over a channel.

# Role of each Member:

1. Harsh Narain Mathur gathered the necessary requirements and did the analysis of the project.
2. Harshit Chauhan worked on the development of the algorithm.
3. Pranay Mahajan worked on the designing of the pseudocode and UML diagram of the project.

**Approved By**

**Mr. Ankit Vishnoi Dr Monit Kapoor**

**Project Guide Head of Department**