## horizontal line



Remote Phonebook

12.10.2022-17.10.2022

**Sprint 2**

Group 6

# 

# INDEX

| **SL. NO.** | **CONTENTS** | **PAGE NO.** |
| --- | --- | --- |
| 1 | Overview. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 2 |
| 2 | Goals. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 2 |
| 3 | Purpose. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 2 |
| 4 | Target audience. . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 2 |
| 5 | Design overview. . . . . . . . . . . . . . . . . . . . . . . . . . . . .  DFD Level 0. . . . . . . . . . . . . . . . . . . . . . . . . . . . .  DFD Level 1. . . . . . . . . . . . . . . . . . . . . . . . . . . . .  Flowchart for remote phonebook. . . . . . . . . .  Flowchart for user. . . . . . . . . . . . . . . . . . . . . . . .  Flowchart for authenticated user. . . . . . . . . . .  Flowchart for admin authenticated user. . . . | 3  3  4  5  6  6  7 |
| 6 | System architecture. . . . . . . . . . . . . . . . . . . . . . . . . .  Functions. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . | 8  8 |
| 7 | Tools report. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  Gcov report. . . . . . . . . . . . . . . . . . . . . . . . . . . . .  Gprof report. . . . . . . . . . . . . . . . . . . . . . . . . . . .  Splint report. . . . . . . . . . . . . . . . . . . . . . . . . . . . .  Valgrind report. . . . . . . . . . . . . . . . . . . . . . . . . . | 11  11  12  13  14 |
| 8 | Testing report. . . . . . . . . . . . . . . . . . . . . . . . . . . . . . .  Unit testing report. . . . . . . . . . . . . . . . . . . . . . .  Integration testing report. . . . . . . . . . . . . . . . . | 15  15  15 |
| 9 | Requirement Traceability Matrix. . . . . . . . . . . . . . . | 18 |

# 

# Overview

**Remote Phonebook for Client :**

In this project ,We have to make a setup for a remote phonebook. The process starts by establishing a connection between the client and the server.User is then authenticated by the use of the pre registered user data stored in an appropriate data structure. The server maintains a concurrency control for all its clients with proper security and protection. The client would ask authentication from the server and after authentication they would remain in the public group by default. There would be three types of access categories for the user: anonymous, authenticated and admin authenticated. For an anonymous client, the server should support a virtual user with the name anonymous to add contact to a public group only without any password. All the authenticated users should be allowed to add, view and delete contact from the phone directory for the group they belong to. They should view the content of public groups. Users with authenticated admin access can access any contact in addition to add, remove group to phone directory.

# Goals

This project aims at creating and maintaining a remote phonebook for a client having a required set of features. It aims at smooth functioning and connections between the client and server. All clients should be able to access certain functions based on their level of authentication.

# Purpose

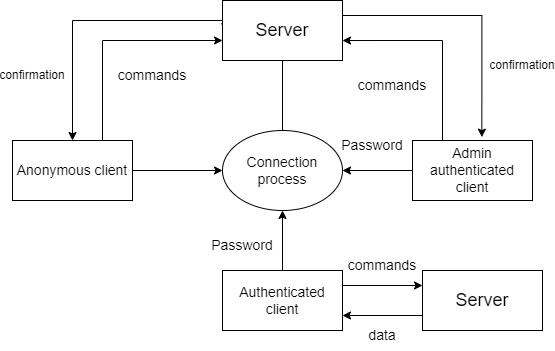
The purpose of this document is to track and record all the information and events occurring throughout the phonebook and keep the details organized so that we can track the progress and functionalities of the clients and the server properly.

# Target audience

The target audience are the clients whose access rights are tracked and functions are specified. Also, the server who has to authenticate the clients and concurrently manage them.

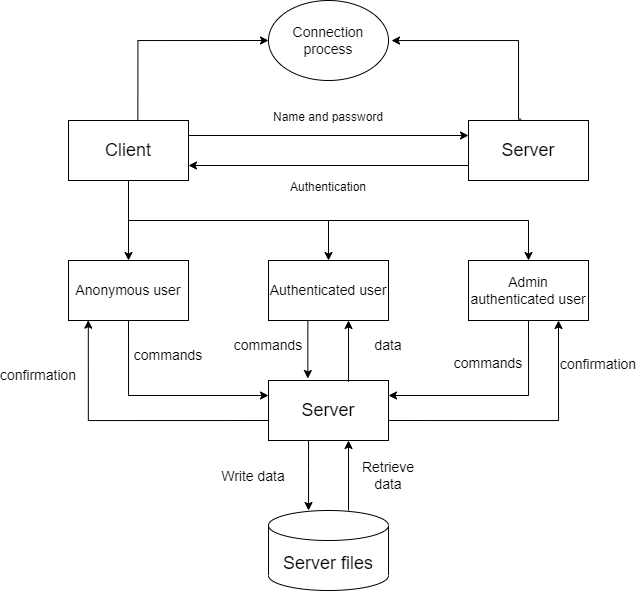
# Design Overview

## Data Flow Diagram Level 0:



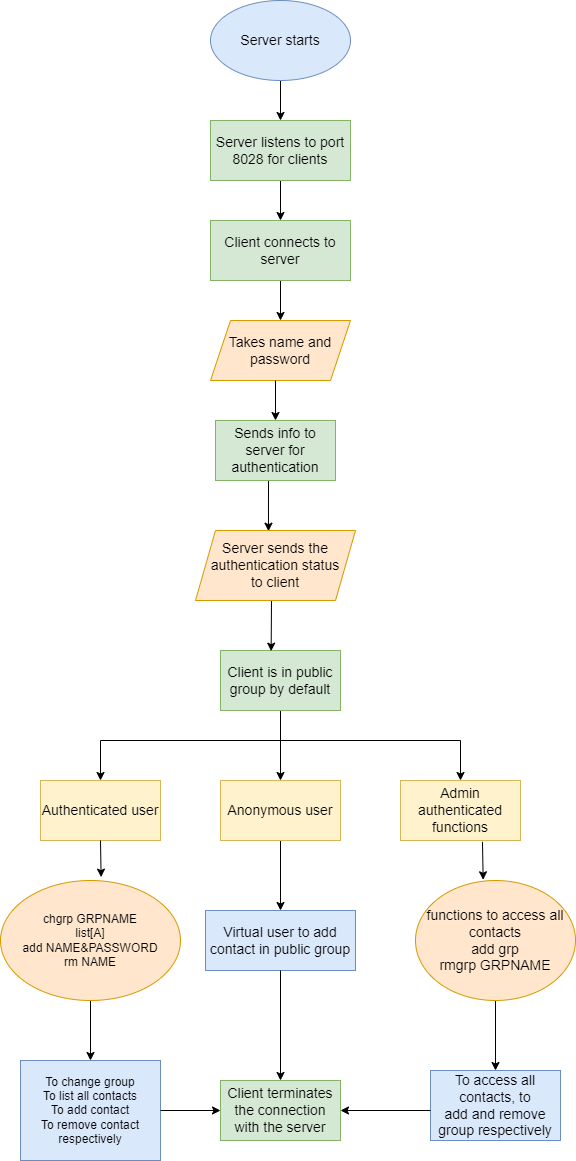
## 

## Data Flow Diagram Level 1:

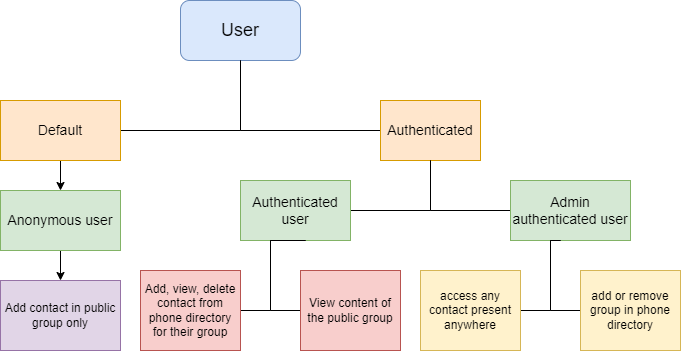


## 

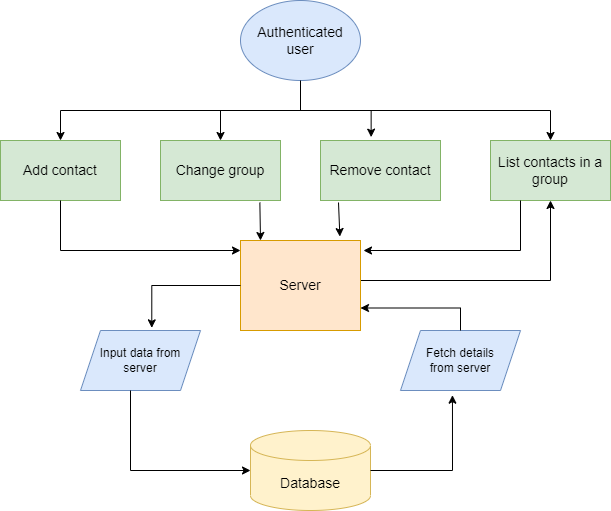
## Flowchart for Remote phonebook:



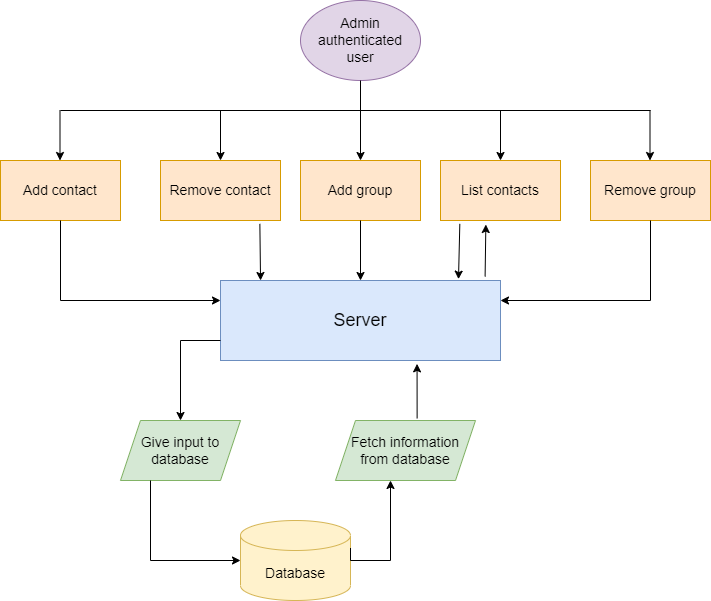
## Flowchart for user:



## Flowchart for functions of authenticated user:



## Flowchart for functions for admin authenticated user:



# 

# System Architecture

## Functions:

### SERVER FUNCTIONS:

**User Authentication:**

The server should authenticate the user from a poll of registered user data kept with appropriate data structure.

**Server side Firewall Protection:**

The server should listen to port 8028 for client connection and make the client side computer access the server port.

**Service starting at boot:**

The server must configure to start the service automatically at boot.

**Server side concurrency:**

The server should support the concurrent client connection.

### CLIENT FUNCTIONS:

**Client side program starting:**

The user should start the client program to connect the Server whenever he wants.

**Client side connection request:**

Client should request a connection to the server on port 8028.

**Client side authentication:**

Client program should start with an authentication

**Client side user environment:**

On successful authentication the user should be placed in a public group for phone book access.

**Client side environment customization:**

Users can use chgrp GRPNAME to change his Working group on phone book.

**Client side browsing contact:**

list[A] to list out all contacts in a group

**Client side contact Add:**

User adds contact using this function.

**Client side contact deletion:**

User can remove any contact from working group

**Client side Group Add and Remove:**

Admin authenticated user use this to add and remove groups.

**Client side quit:**

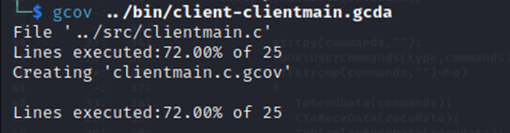
The user uses the subcommand bye to terminate the connection.

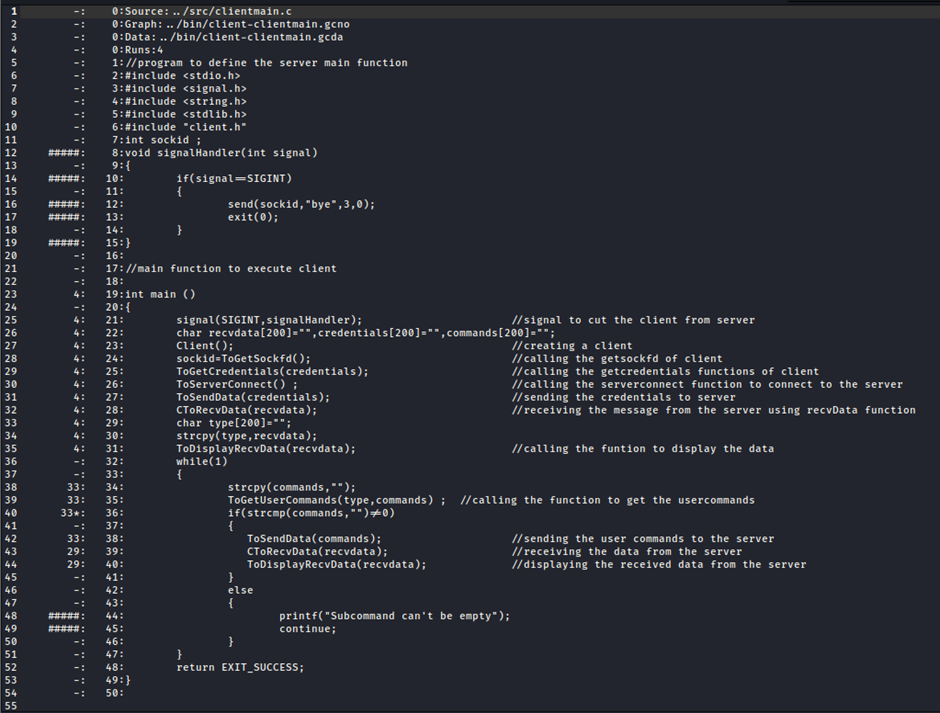
# 

# 

# Tools Report

## Gcov report:



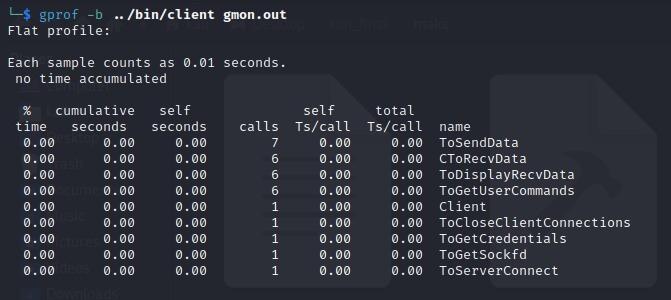


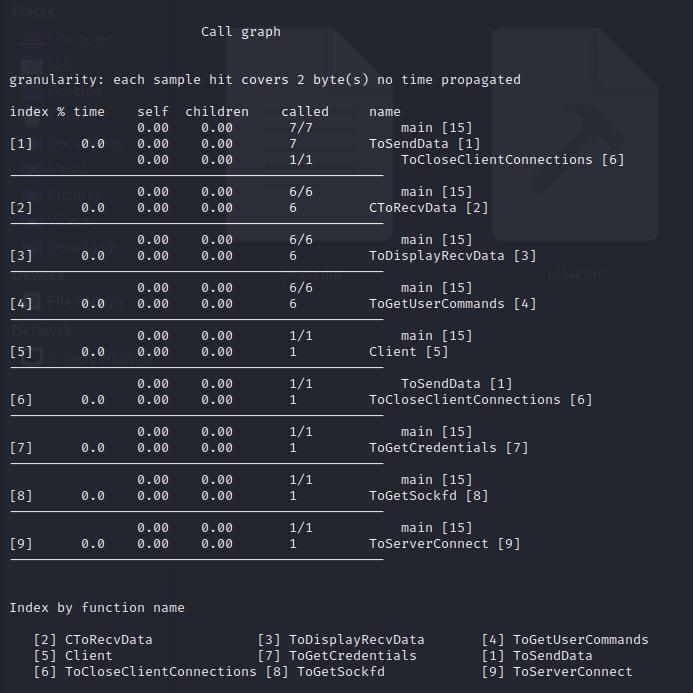
## 

## 

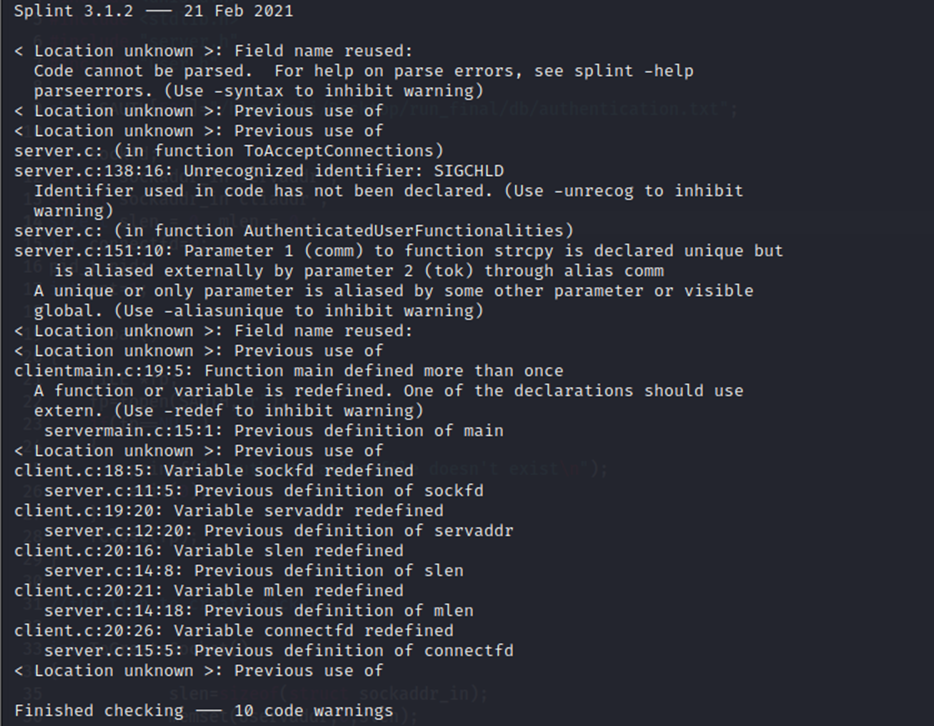
## 

## Gprof report:





## Splint report:



## 

## 

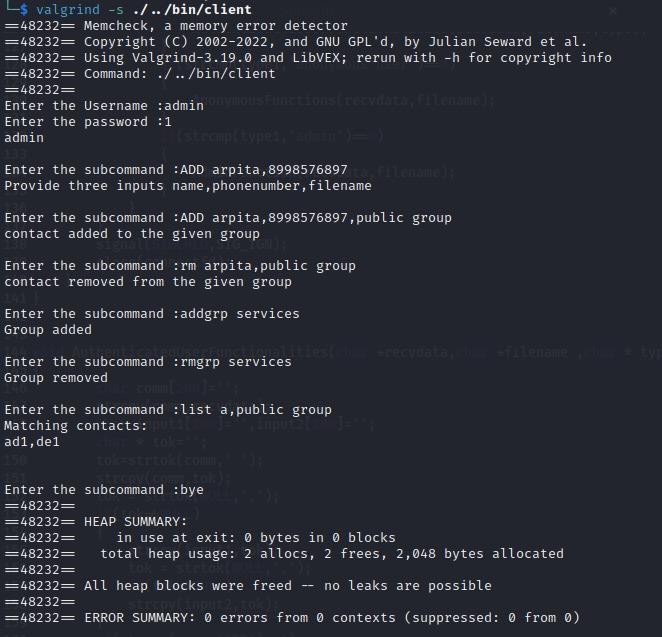
## 

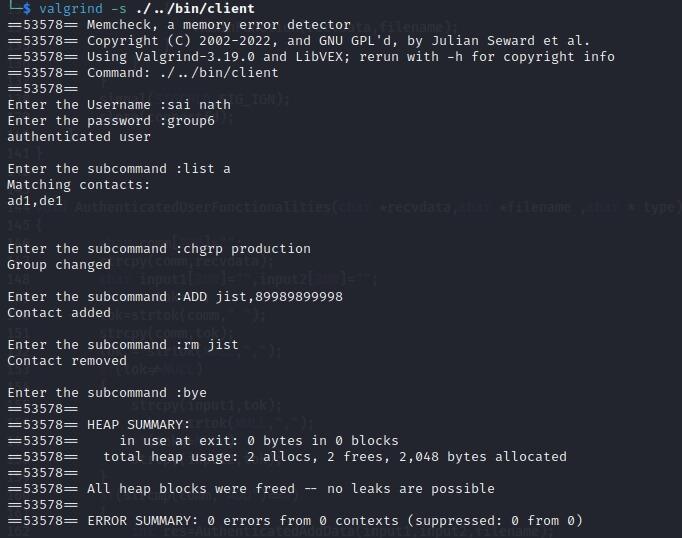
## 

## 

## 

## Valgrind report:



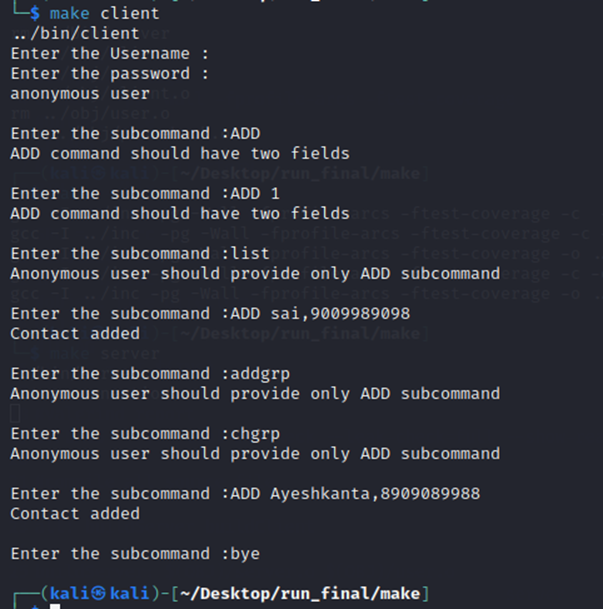


# Testing Report

## Unit testing report:

## Integration testing report:

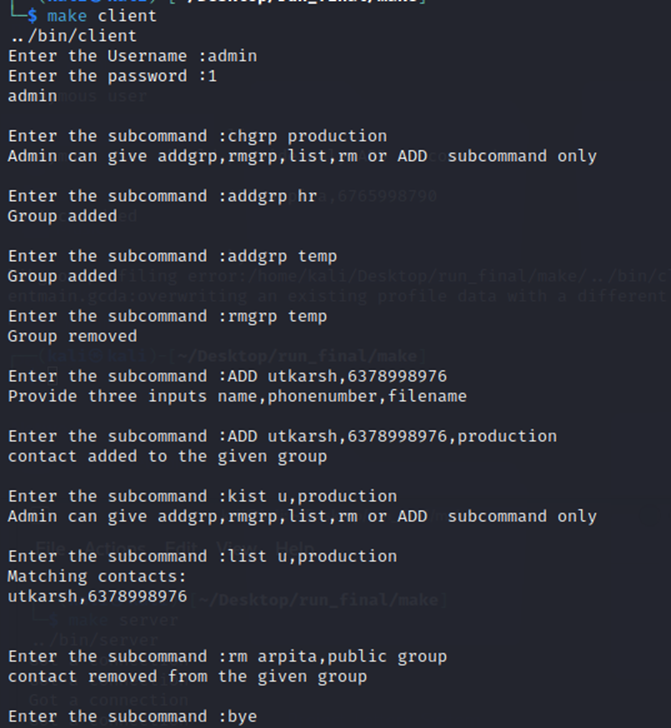
Case 1:Anonymous user



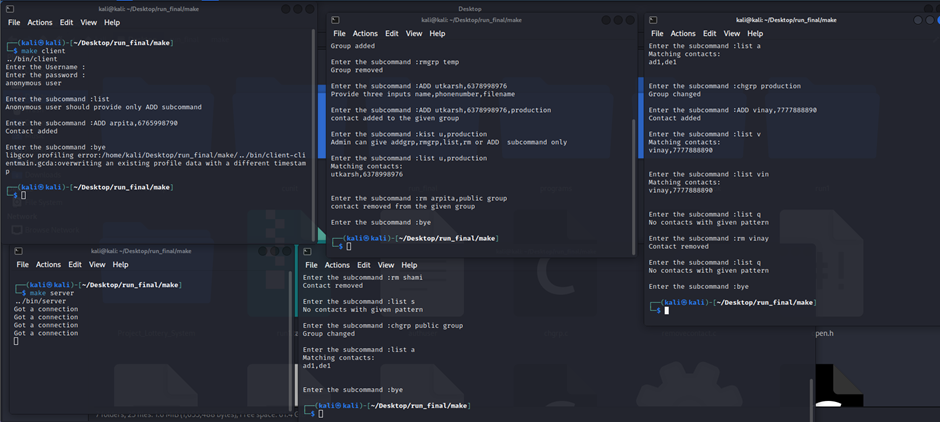
Case 2:Authenticated user



Case 3:Admin authenticated user



Case 4: Handling multiple clients



# Requirement Traceability matrix(RTM):

| **REQUIREMENT** | **DESIGN MAPPING** | **CODE MAPPING** | **IT MAPPING** | **UT MAPPING** |
| --- | --- | --- | --- | --- |
| RPBC\_01 | a | Client-server connection |  |  |
| RPBC\_02 | b | User authentication |  |  |
| RPBC\_03 | c | Server side protection |  |  |
| RPBC\_04 | d | Anonymous user access | IT\_01 |  |
| RPBC\_05 | e | Authenticated user access | IT\_02 |  |
| RPBC\_06 | f | Authenticated admin access | IT\_03 |  |
| RPBC\_07 | g | Server side concurrency | IT\_04 |  |
| RPBC\_08 | h | Client side program starting |  |  |
| RPBC\_09 | i | Client side connection request |  |  |
| RPBC\_10 | j | Client side authentication |  |  |
| RPBC\_11 | k | Client side user environment |  |  |
| RPBC\_12 | l | Change group |  | UT\_03  UT\_04 |
| RPBC\_13 | m | Contact list display |  |  |
| RPBC\_14 | n | Add contact |  |  |
| RPBC\_15 | o | Contact deletion |  | UT\_01  UT\_02 |
| RPBC\_16 | p | Group add and remove |  |  |
| RPBC\_17 | q | Client side quit |  |  |