GROUP – 5

CALL DATA RECORD

**Overview of the CDR System**

|  |
| --- |
| * CDR specifies charging functionality and charging management for cellular service providers. * Handle the CDR files that contain CDR records. * Each CDR record contains descriptions of charging events such as voice calls and SMS messages with all relevant data. * **CDR File Format**:   The CDR files contain a header section followed by a variable sized CDR data section. The CDR data section contains zero or more concatenated CDR records. Each CDR record is encoded on a single line.   CDR record is composed of the following fields separated by |  • MSISDN: unique number identifying a subscription in a mobile network maximum length of an MSISDN to 7 digits.  • Operator Brand Name: a string maximum 64 chars. Such as “Airtel”.  • Operator MMC/MNC: mobile country code and mobile network code maximum 6 digits.  • Call type: call type one of:  - MOC: outgoing voice call.  - MTC: incoming voice call.  - SMS-MO: outgoing message.  - SMS-MT: incoming message.  - GPRS: internet  • Duration: in seconds  • Download: MB downloaded if type is GPRS  • Upload: MB uploaded if type is GPRS  • Third party MSISDN: corresponding third party in this call, empty for GPRS  • Third party operator MMC/MNC: corresponding third party’s mobile operator in this call. |

**Key Features**

* Multiple client login
* User credentials stored in a file.
* Large data compatible
* Easy to understand function names
* Data gets saved to file functionality

**--> How the System Works**

1. **Client Login/Signup**

User Account Management: The system allows for easy user management where new users can create an account through a signup process.

Authentication: Existing users can log in using their credentials. The authentication process ensures that only authorized users can access sensitive functionalities, enhancing the system's security.

2. **Process CDR File**

CDR Processing: Upon receiving the file, the system employs robust algorithms to read and parse the data. This process includes:

Customer Billing: The system extracts and aggregates data specific to each customer, allowing for accurate billing based on usage.

Inter-Operator Settlement: The data is further aggregated to manage settlements between different telecom operators, ensuring fair charges and compensation for services rendered.

3. **Store Processed Data**

Data Storage: The processed data is stored in text files for ease of access and retrieval:

CB.txt: Contains all information related to Customer Billing.

IOSB.txt: Contains data relevant to Inter-Operator Settlements.

**--> Getting Started**

* **Structure of files**

**sprint directory**--> server directory: Makefile

**Inside server directory:**

* **include directory - ->** server.h, billing.h, user\_management.h
* **src - ->** billing.c, main.c, server.c, user\_management.c
* **data - ->** user.txt, users.txt
* **bin - ->** serv

**sprint directory** --> client directory: Makefile

**Inside client directory:**

* **include directory - ->** client.h, client\_operations.h, network\_utils.h
* **src directory - ->** client.c, client\_operations.c, network\_utils.c
* **data directory - ->** user.txt, users.txt
* **bin directory - ->** client

Follow the steps below to get the system up and running:

* Install Putty server from Microsoft Store and login with the credentials.
* Install Dependencies: Make sure the necessary libraries for socket programming, multithreading, and file handling are installed.
* Libraries: Socket Programming (<sys/socket.h>, <arpa/inet.h>, <unistd.h>), Multithreading (<pthread.h>), File Handling
* Go in the sprint folder by using command: cd sprint.

**How to run the software?**

* Start the Server:
* Run the server program that listens for client connections and processes their requests.

Compile Command: gcc new\_serv.c -o new\_serv -lpthread

Run Command: ./new\_serv

* Run the Client:

Once the server is running, start the client program to interact with the system.

Compile Command: gcc new\_client.c -o new\_client

Run Command: ./new\_client

**System Requirement:**

Operating System: Windows/Linux/MacOS  
Languages: C Programming Language  
RAM: Minimum 2 GB  
Storage: 10 MB for storing CDR files

**Configuration:**

Server Configuration: Ensure the server is configured to run on port 8080. Modify the port and IP address if required in the source code.  
Client Configuration: The client connects to the server using the server's IP address and port. Modify the IP address in the client code to match the server configuration.  
CDR File Format: Ensure that CDR files follow the specified format with fields separated by |.

**Conclusion:**

The CDR System efficiently processes large telecom data records, ensuring accurate customer billing and inter-operator settlements.