Client-Server Database

Low-Level Design Version Draft v7

Document Control:

Project Revision History					
Date	**Version**	**Author**	**Brief Description of Changes**		

Team Members	
Employee ID	**Name**
46290158	**Muskan Pathan Allabaksh**
46282451	**Sri Harshita Talari**
46282452	**Renuka Gujjala**
46282098	**Bhagyalakshmi Daliparthi**
46282102	**Sarika Pathiputturu**
46290159	**Vinitha Devalla**

Table of Contents

I) Low-Level Design

Low-Level Design

1. Introduction

The aim of this document is to gather, analyze and give an in-depth overview of the Project i.e., Data-Base Server. It is an application to store and retrieve different types of data (e.g., Employee data, School data, etc..). A list of supported data types and format is published by the server. All clients connecting to server will send data to store in the server and they can retrieve the data from Server. The Database Server is based on the concept of communication between client and server and managing the data in database. The server will maintain the different type and formats of data in database and will receive the relative data from client and will add to the database. The server can connect to single client at a time. No two clients can access or modify the same database at a time. The server should employ proper connection of database in case two clients attempt to modify the database values at same time.

1.1 Purpose

The purpose of this document is to describe the low-level design flow of the current project description to represent a suitable model for coding.

1.2 Document Conventions

TBD (To be continued).

1.3 Intended Audience and Reading Suggestions

This is general-purpose Software Thus any one Can Access it.

1.Client
2.Development Team
3.Maintenance Team

The document is primarily intended for team members, which consists of trainees under the **Capgemini** Training Program.

1.4 Refers

The references are:

[1] System Requirements Specification Document
[2] System Specification Requirement

2. Detailed System Design

2.1 Design Descriptions:

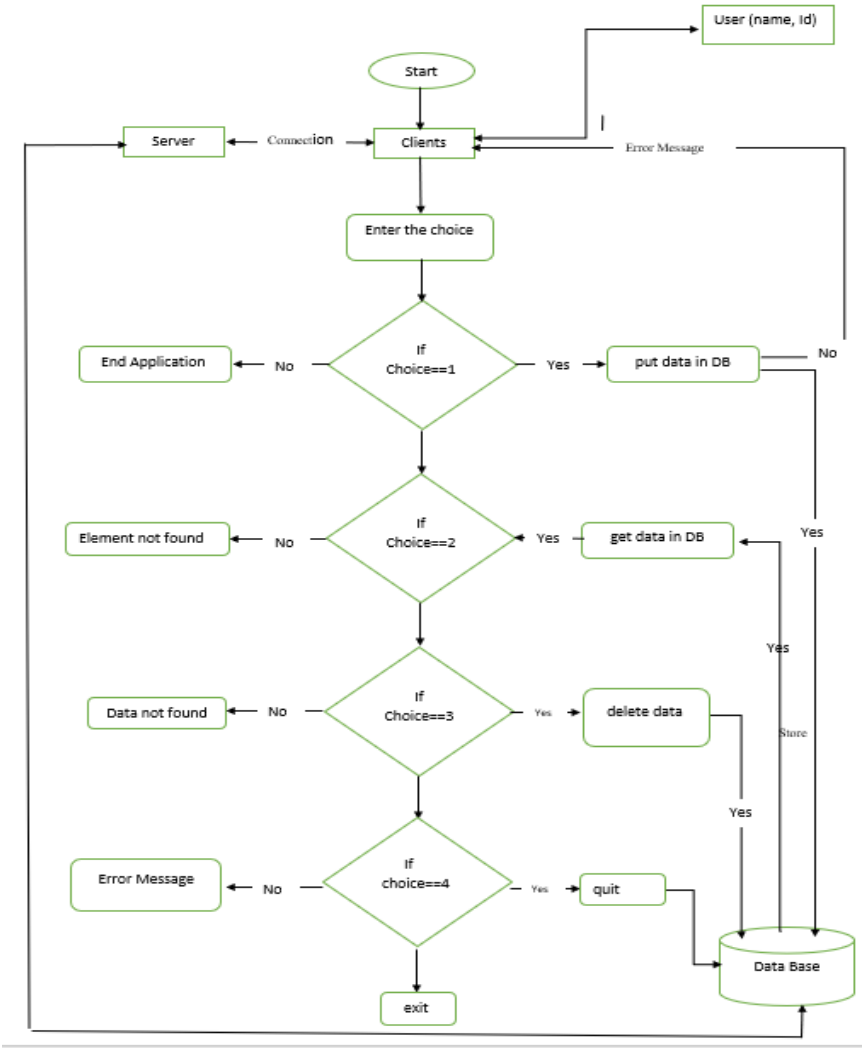
This project design mainly focuses on implementing of Client Server model. User will be able to choose the based on reequipment, which they want to implement. In these projects after the successful connection of Server and Client, the Server will display the Database format to user. Users have to select a choice of Database in which, he wants to add, delete, retrieve and display the data. Based on the choice the user has entered the algorithm will continue the process.

Here, the Client-Server and Database will help in understanding the storage of data in database and commination between them.

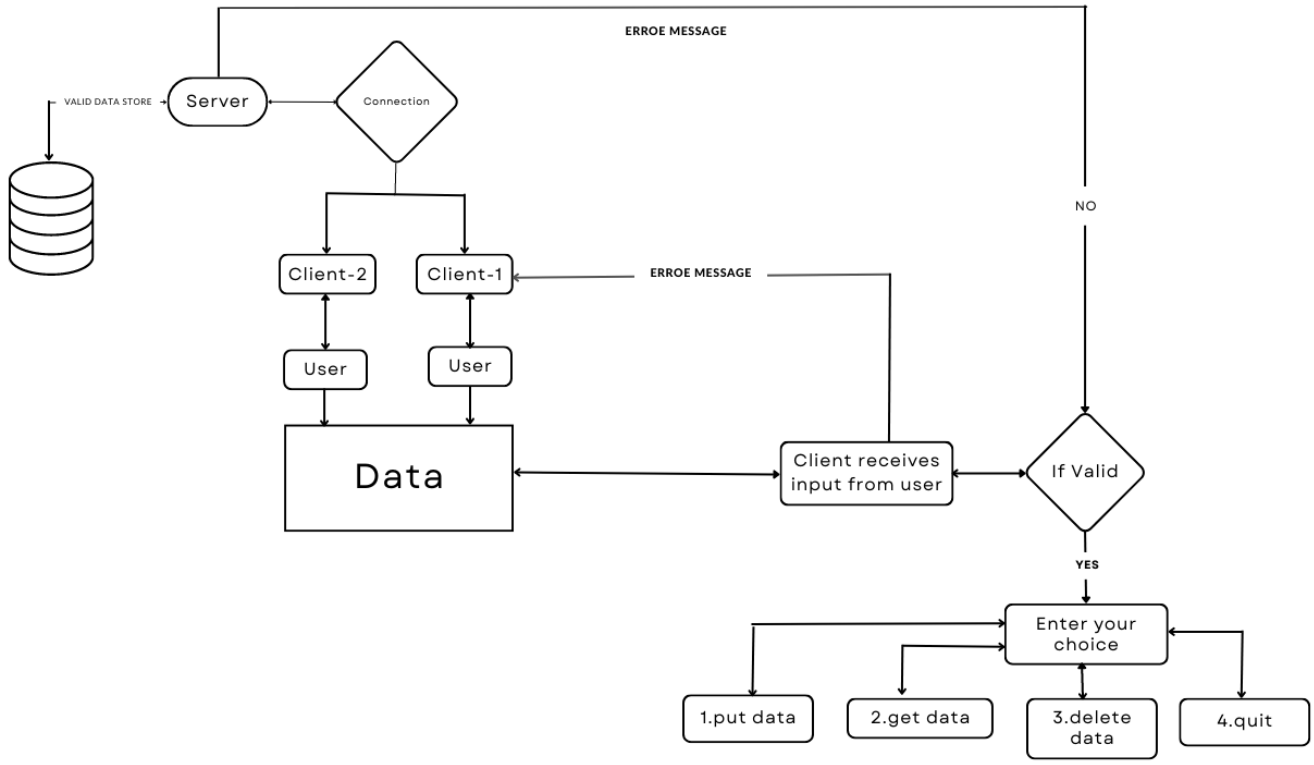
Main menu:

User gets options selection menu or to exit the program.

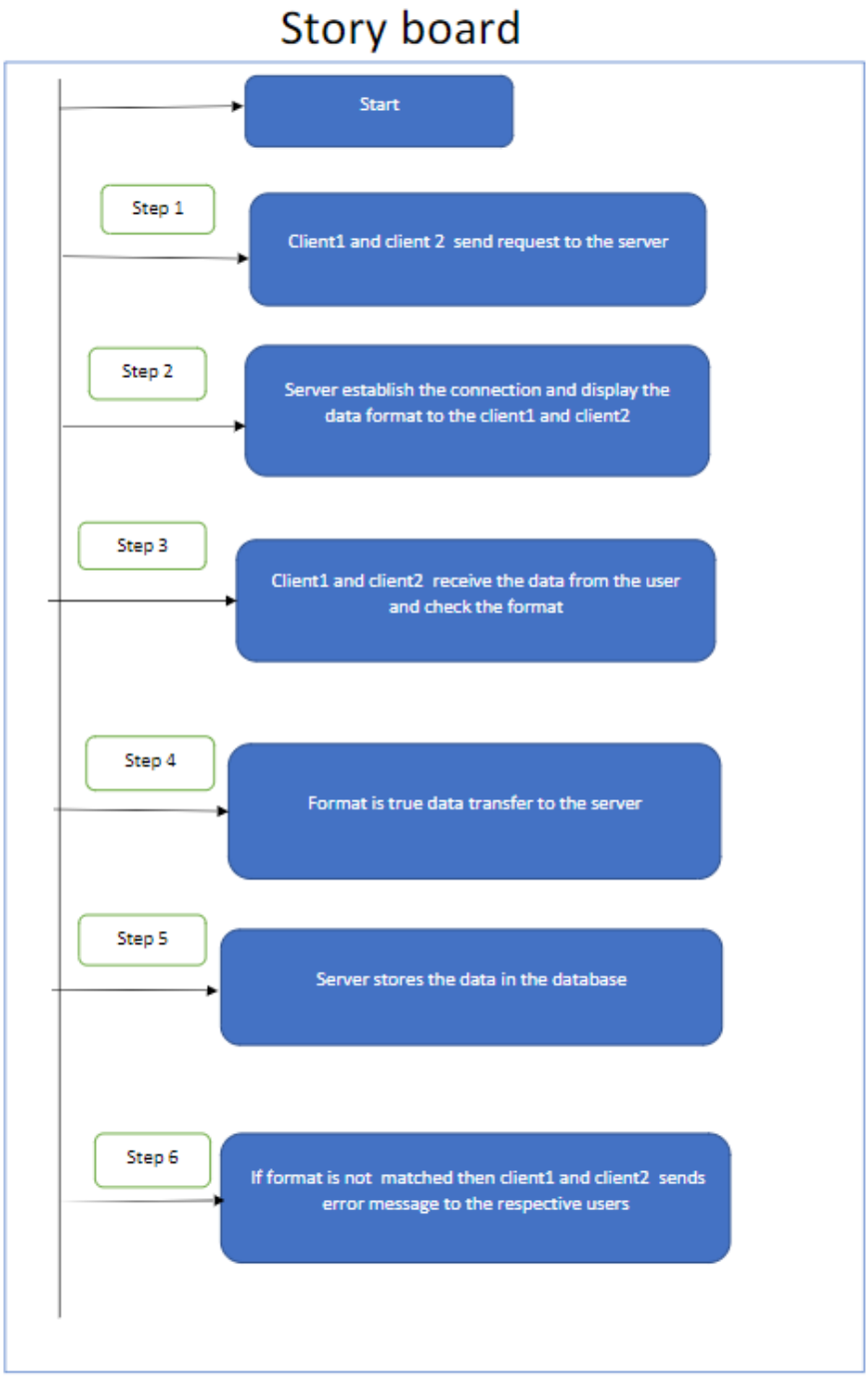
2.2 Flowchart



2.2.2 Menu level 2



2.2.3 Storyboard


Story board
Start
Step 1
Client1 and client 2 send request to the server
Step 2
Server establish the connection and display the data format to the client1 and client2
Step 3
Client1 and client2 receive the data from the user and check the format
Step 4
Format is true data transfer to the server
Step 5
Server stores the data in the database
Step 6
If format is not matched then client1 and client2 sends error message to the respective users

2.3 Modules

2.3.1 dbclient.c

main()

Name	main			
Input	Parameter Name	char, int	Initial Value: NA	To check whether user giving right input
Output	Return value type	int	-	-
Description	The main function will perform the socket creation and connection to the server. It will display the menu driven interface to the user.			
Pseudo Code	1. Display the main menu 2. Wait for user options 3. Based on user input perform particular action			

2.3.2 connectServer()

Name	connectServer			
Input	Parameter Name	char, int	NA	-
Output	Return value type	int		NA
Description	The function is used to connecting the client to server using hostname and port number			
Pseudo Code	1. create socketfd 2. connect to server			

2.3.3 startData()

Name	startData			
Input	Parameter Name	int	Initial value: NA	-
Output	Return value type	int	-	-
Description	This function will handle the database interactions			
Pseud Code	1.If User chooses option 1: PUT will be done 2.If User chooses option 2: GET will be done 3. If User chooses option 3: DELETE will be done 4.If User chooses option 4: QUIT will be done			

2.3 Modules

2.3.1 dbserver.c

main()

Name	main			
Input	Parameter Name	Int,char	Initial value: NA	To check whether user gives the right input
Output	Return value type	int		-
Description	The main function will perform the socket creation and connection to the server.			
Pseud Code	1.start the server			

2.3.5 startServer()

Name	startServer			
Input	Port	int	Initial value:NA	
Output	Return value type	int	-1 if fail	-
Description	It will bind and accept the client request to establish the connection			
Pseud Code	1.create socketfd 2. perform bind 3. listen to the client request 4. create threads to handle client request			

2.3.6 manageClient()

Name	manageClient			
Input	Parameter Name	void	Initial value: NA	-
output	Return value type	-	-	-
Description	It will handle the operations requested by multiple clients			
Pseud Code	1. Create a database 2. If client send data, store that to database 3. If client request to get data, will send data to client 4. If client want to delete the data, it will get delete			

2.3.7 setIndex()

Name	setIndex			
Input	databasefd	int	Initial value: NA	-
Output	Return	-		-

	value type			
Description	This function is used to find the next open spot in database to write the data			
Pseud Code	1. Set the index at starting of the database 2. write the data			

2.3.8 findID()

Name	findID			
Input	databasefd, id	int, int	Initial value: NA	-
Output	Return	Buffer	Initial value: NA	
Description	This function will check for the id whether it is present or not			