Remote Backup Utility

Software Requirement Specification (SRS) Document

INDEX

Application features and requirements **4**

Application Features **4**

Application process **4**

Required Tools **5**

Data Flow Diagram **5**

DFD Level 0 **5**

DFD Level 1 **5**

Description **3**

System Requirement **4**

Flow Diagram **6**

Introduction 3

Purpose 3

Scope 3

1. Introduction

*“*Remote Backup Utility” main aim is to create the remotely controlled backup making program of client files which are stored to server side and many different backups is made by this program.

* 1. Purpose

The purpose of this document is to show the requirements for the Remote Backup Utility application, in which Application to be developed for backing up files to server side from client side.

* 1. Scope

This project aims to create the development of Remote Backup Utility application. User starts the program and connects to server through Socket Ports and then program shows a menu to user that which type of backup user want. In this program, there are 4 types of backup user can do with files. “Full Backup”, “Incremental Backup”, “Scheduled Backup”, “Just in Time Backup”.

1. Description

It is an application which is used to create and store backups for user.

In this TCP/IP Socket programming is used to connect client side to server side. It provides the routines required for interprocess communication between applications, either on the local system or spread in a distributed, TCP/IP based network environment. Once a peer-to-peer connection is established, a socket descriptor is used to uniquely identify the connection. The socket descriptor itself is a task specific numerical value.

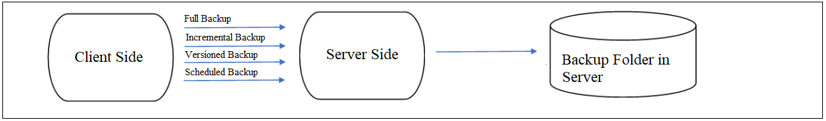
Socket programs are used to communicate between various processes usually running on different systems. It is mostly used to create a client-server environment. This post provides the various functions used to create the server and client program and an example program. In the example, the client program sends a file name to the server and the server sends the contents of the file back to the client.

* 1. System Requirements
* System should have Ubuntu Linux installed
* System should have either 4GB or more RAM
* The service is used preferably on a desktop or laptop

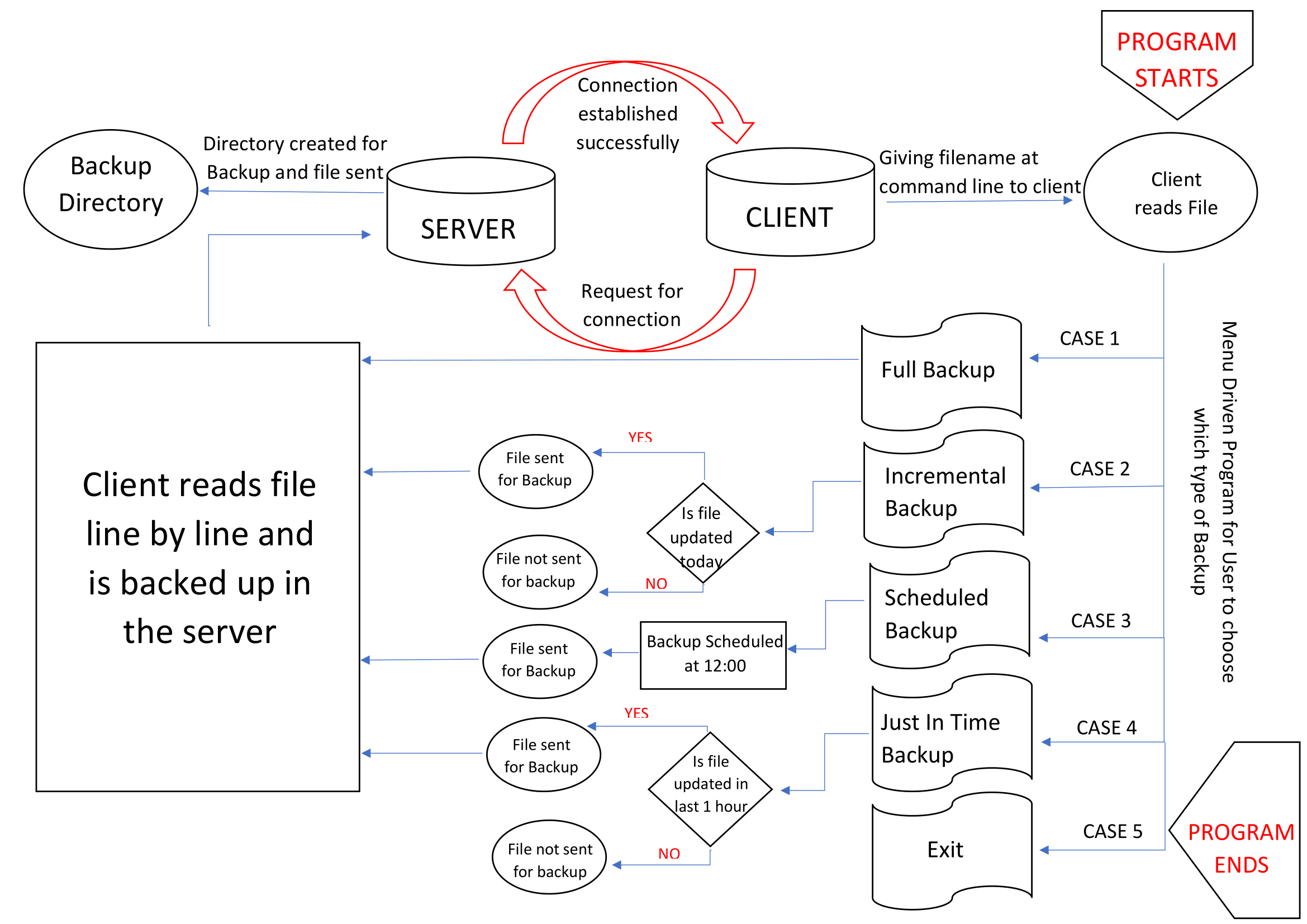
1. Application features and requirements
   1. Application process

* RBU\_01 -> **send\_file**: We create this function taking the single file as argument from client and storing it on server-side.
* RBU\_02-> **send\_file\_multiple**: We can also take multiple files as argument on terminal from user and store all files to server side.
* RBU\_03-> **Increment\_Backup**: We created this function as it can take same file for backup only if it has been modified at same day of backup and store that backup to server-side.
* RBU\_04-> **Increment\_Backup\_multiple**: As this works same as Increment\_Backup but with multiple files support just need of multiple files as arguments from user.
* RBU\_05-> **Scheduled\_Backup**: This function helps user to backup any file at specific time pre-given. So user has to drop the file and backup will be automatically stored on server at specific time.
* RBU\_06-> **Scheduled\_Backup\_multiple**: This code works as same only the change of getting multiple files as argument from user is needed and automatically they will be backed up on server-side at specific time.
* RBU\_07-> **Just\_In\_Time\_Backup**: This code take file from a folder with multiple files, that file which is modified last 1 hour by user is backed up to server-side.
* RBU\_08-> **Just\_In\_Time\_Backup\_Multiple**: This function takes multiple folder from user and find those files that are modified by user and create backup to server-side of them.
  1. Required Tools
* C Language
* System Programming
* Socket Programming
* C Unit Library
* C File Handling
  1. Application Features
* **Accessibility**: The system is easy to access.
* **Language**: The system is built using System and C language
* **Capability**: Remote Control Backup Utility can replace the high cost backup services which has very less security. This program is properly secured as it can only be used in Linux by user and access their items anywhere anytime
* **Reliability & Availability**: The system is available 24/7 that is whenever the user would like to use the system, they can use it up to its functionalities.
* **Performance**: The system will work on the user’s terminal.

1. Data Flow Diagram:
   1. DFD Level 0 -

****

* 1. DFD Level 1 -



1. Flow Diagram

