

Project Report (Major Project- II)

on

Personal Mobile Storage Sharing

Submitted in partial fulfillment of the requirements

for the award of the degree of

Master of Technology

in

Software Technology

By

Nitesh Suthar

Roll No.: - 2K16/SWT/509

Under the guidance of

Dr. Rajesh Kumar Yadav

Assistant Professor



Department of Computer Science & Engineering

Delhi Technological University

(Formerly Delhi College of Engineering)

Bawana Road, Delhi 110042

2020



Delhi Technological University
(Formerly Delhi College of Engineering)
Bawana Road, New Delhi-42

DECLARATION

I hereby declare that the thesis entitled “**Personal Mobile Storage Sharing**” which is being submitted to the Delhi Technological University, in partial fulfilment of the requirements for the award of the degree of Master of Technology in Software Technology is an authentic work carried out by me. The material contained in this thesis has not been submitted to any university or institution for the award of any degree.

DATE: 29-06-2020

Nitesh Suthar

SIGNATURE:

NITESH SUTHAR

2K16/SWT/509

CERTIFICATE



Delhi Technological University

(Formerly Delhi College of Engineering)

Bawana Road, New Delhi-42

This is to certify that project report entitled “**Personal Mobile Storage Sharing**” done by me for the Minor Project II for the award of degree of Master of Technology Degree in Software Technology in the Department of Computer Science & Engineering, Delhi Technological University, New Delhi is an authentic work carried out by me.

Signature :

Nitesh Suthar

Student Name : **Nitesh Suthar**

Roll No. : **2K16/SWT/509**

Above Statement given by Student is Correct.

Signature : *Dr. R.K. Yadav*
Project Guide : **Dr. Rajesh Kumar Yadav**

Assistant Professor

Department of Computer Science & Engineering

Delhi Technological University

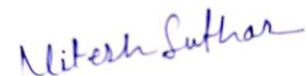
Acknowledgement

No volume of words is enough to express my gratitude towards my guide **Dr. Rajesh Kumar Yadav**, Department of Computer Science & Engineering, Delhi Technological University, Delhi, who has been very concerned and has aided for all the materials essentials for the preparation of this project report. He has helped me to explore this vast topic in an organized manner and provided me all the ideas on how to work towards a research-oriented venture.

I am also thankful to **Dr. Rajni Jindal**, HoD of Computer Science & Engineering Department and **Dr. Ruchika Malhotra**, Coordinator, for the motivation and inspiration that triggered me for the project work.

I would also like to thank the staff members and my colleagues who were always there at the need of hour and provided with all the help and facilities, which I required, for the completion of my project work.

Most importantly, I would like to thank my parents and the almighty for showing me the right direction, to help me stay calm in the oddest of the times and keep moving even at times when there was no hope.



Nitesh Suthar

(2K16/SWT/509)

Abstract

In this era of Mobile devices, every user is willing to use mobile devices, which are capable of providing high performance, and which can provides more storage spaces. People are getting aware about using various cloud-based applications provided by network operator and service providers. Among those applications, there are multiple cloud-based usability's which provide the access to the Cloud Storage. There are many cloud-based application available which provide private storage, such as Microsoft OneDrive, Google Cloud, Dropbox, etc. All of them store date on cloud servers. Even though these are secured still data is exposed to service provider so there is always a risk of data being stolen by some hackers or intruders.

We are living in the era of smartphones, every user is having a smart phone with different storage capabilities such as 8, 16, 32, 64 and 128 GB of space and even more storage is going to be embedded in near future. We always find some user, which are not using even 50% of total available storage in their smartphone, which remains unused throughout the lifetime of Mobile device. Therefore, in this Major Project – II, we propose a new framework, which provides a system where a user can share his device's free or unused storage space with other users. Consider a family; where we have n members with n different mobile devices then the aggregated free storage space of all n number of mobile phones can be used as common private storage.

The proposed system is called **Personal Mobile Storage Sharing (PMS-Sharing)**. PMS-Sharing enables user to form a local group of users and allow them to access shared storage directory. The system removes dependencies on cloud storage service provider to store the data on servers in order to access it anywhere around the globe. User's data is stored on the devices within the known group of family members, by this way his data is safe and reachable physically in case network is not available due to some natural hazard. PMS-Sharing will be very useful application for the military, secure forces, corporate network, and for those users who do not want to upload their personal data on cloud-storage service and user does not know physical location of storage.

Contents

Acknowledgement.....	4
Abstract	5
Contents.....	6
List of Tables.....	7
List of Figures	8
Chapter 1 - Introduction.....	10
1.1 What is Cloud Storage?.....	10
1.2 Cloud Storage Architecture	10
1.3 Types of Cloud Storage Systems.....	11
1.3.1 File Storage System.....	11
1.3.2 Block Storage System.....	11
1.3.3 Object Storage System	11
1.4 Background and Motivation.....	11
1.5 Scope & Objectives	12
1.6 Thesis Outline.....	13
Chapter 2 Literature Review.....	14
2.1 Cloud Storage System	14
2.2 Review of Cloud Storage Providers	15
2.3 Advantages/Benefits of using Cloud Storage.....	17
2.4 Disadvantages/Challenges of using Cloud Storage	18
2.5 Existing Cloud Concept.....	18
2.6 Motivation & Problem Statements	19
2.7 Current Challenges and Limitations.....	20
2.7.1 Battery Consumption.....	20
2.7.2 Data recovery in case of loss of device.....	20
2.7.3 Connectivity.....	20
Chapter 3 Proposed Work.....	21
3.1 Overall Description of PMS-Sharing	21
3.2 Perspective of PMS-Sharing.....	23
3.3 Product Functions.....	24
3.4 User Classes and Characteristics	24
3.5 Operating Environment	25
3.6 Design and implementation constraints.....	25

3.7 Feature Specification	25
3.7.1 User Interface	25
3.7.2 Hardware Interface	41
3.7.3 Software Interface	41
3.7.4 Communication Interface	41
Chapter 4 Implementation, Setup and Analysis	43
4.1 PMS-Sharing Architecture	43
4.2 Algorithm for Key Function and Features.....	46
4.2.1 Algorithm for Group Formation	46
4.2.2 Algorithm to display Remote/Local Directory View	49
4.2.3 Algorithm for Device Identification & User Authentication.....	50
4.3 Database Implementation	54
4.3.1 User Repository	54
4.3.2 File Repository	55
4.4 Setup, Evaluation and Analysis	55
4.4.1 Prerequisite	55
4.4.2 Setup PMS-Sharing application	55
4.4.3 Evaluation and Analysis based on case studies	56
Chapter 5 Conclusion and Future Scope.....	59
5.1 Future Work	59
References	60

List of Tables

Table 1: Differences between File, Block and Object Storage System.....	15
Table 2: Review of popular Cloud Storage Applications.....	17
Table 3: Challenges of using Cloud Storage	18
Table 4: Input Constraint on User Signup Page	29
Table 5: Evaluation using Case Study#1	57
Table 6: Evaluation using Case Study#2.....	58

List of Figures

Figure 1: General Cloud Storage Architecture	10
Figure 2: Existing Cloud Concept	19
Figure 3: PMS-Sharing Concept	21
Figure 4: Aggregated PMS-Sharing Size	22
Figure 5: PMS-Sharing Use case diagram.....	23
Figure 6: Login Screen	26
Figure 7: Sign Up Screen	27
Figure 8: Login Flowchart.....	28
Figure 9: Forgot password Screen.....	30
Figure 10: Sign-Up Process Flowchart.....	31
Figure 11: Reset password Flowchart	32
Figure 12: Dashboard Screen	34
Figure 13: Dashboard Flowchart.....	35
Figure 14: New Group Screen.....	36
Figure 15: Edit Group Screen.....	36
Figure 16: Local Directory View.....	37
Figure 17: Flowchart to create new group.....	38
Figure 18: Edit Group Flowchart	39
Figure 19: Remote Directory View	40
Figure 20: PMS-Sharing Communication Interface.....	42
Figure 21: PMS-Sharing Architecture.....	43
Figure 22: NSD Module	46
Figure 23: Sequence Chart Group Formation.....	47
Figure 24: Sequence Chart Remote Directory View	50
Figure 25: Generation of message digest on Group Owner Device	52
Figure 26: Decryption of message digest on User Device	52
Figure 27: User Authentication	53

Figure 28: Sequence Chart Remote Directory View 53

Chapter 1 - Introduction

1.1 What is Cloud Storage?

Cloud Storage enables user to store their personal data on remote servers and provides service to save, modify, delete and share their data. User utilizes Cloud Storage services over an internet network. User can access data from anywhere over an internet network. Cloud Storage provider keeps user data on remote servers making available over internet. In Cloud Storage System, data is stored on multiple third-party servers, rather than storing on dedicated server used in traditional network data storage. There are multiple benefits of storing data on Cloud Storage in terms on security, accessibility and of course, no maintenance required from user point of view because Cloud Storage Service Provider companies maintain it.

1.2 Cloud Storage Architecture

Each provides have different way of handling Cloud Storage services. Typical Cloud Storage Architecture consists of Front End User Interface, Middleware protocol and Back End database and file management and Physical Storage. Front End user interface is web server page to access data or file system like view. Middleware provides the actual logic to store, delete, backup, secure and recover data. Back End is actual physical storage and depended hardware.

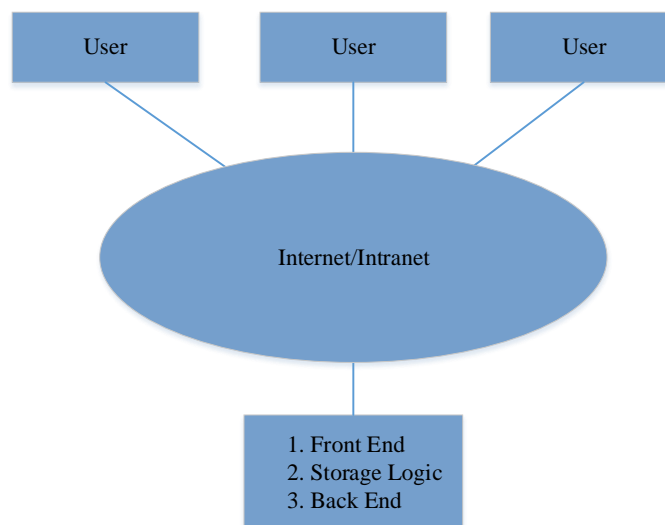


Figure 1: General Cloud Storage Architecture

1.3 Types of Cloud Storage Systems

There are various systems to manage data on cloud servers. Broadly, there are three types of Cloud Storage Systems as explained below and each system has its own advantages and drawbacks.

1.3.1 File Storage System

There are many types of File Storage with different characteristics like structure, access method, speed, security, and size. Both local as well as network storage devices can use File Storage system. Some necessary network file storage are Network File System (NFS), Server Message Block(SMB)/ Common Internet File System (CIFS) and Plan 9 File system protocol (9P)[1][8].

1.3.2 Block Storage System

Block Storage provides a way to store data in evenly-sized memory blocks, used to host the database, which supports random read and write operations[2][8]. Amazon Elastic Block Storage (EBS) is developed for use with Amazon Elastic Compute Cloud (EC2) to deploy a broad category of data loads such as databases, containerized applications, big data analytics engine, file system, media data, and enterprise application data[9].

1.3.3 Object Storage System

Object Storage provides an easy way to store unstructured data in the form of Objects. An object is a simple repository with the unique ID number, metadata, and actual data [3]. Amazon Simple Storage Service (Amazon S3) is capable of managing a significant amount of unstructured data.

1.4 Background and Motivation

With the rollout of 4G and 5G technology, we are assuming that there will be no issue of connectivity and smartphone market is increasing day by day. There are many companies engaged in developing new smartphones with new, improved capabilities with reasonable price. This motivates smartphone users to upgrade their smartphone to the advanced one. Even though many vendors provide exchange offer to buy new phone, still the deal does not go attractive, so the user always thinks to keep that old phone as a backup, and they always choose and prefer to buy new smartphone.

Cloud computing is an emerging model for business computing [5]. In this era of smart mobile devices, millions of users are using smartphones and tablets; these smartphones are equipped with high-performance processors and RAM for computing and numerous gigabytes of inbuilt storage memory along with additional SD card slot to extend the storage capacity. Cloud storage is a new concept extended and developed from the concept of cloud computing [6] and substantial growth of smartphone devices increases the need to use Cloud Storage. In this era of smart mobile devices, every user is having devices with different-different storage capacity i.e. 8, 16, 32, 64, 128 GB and even more storage will be available in next generation of devices. Often we see that user upgrade to new smart phone and their old smart became unused. This motivates us to build a model and framework to build a Personal Mobile Storage Sharing (PMS-Sharing) by aggregating unused free space of any number of unused smart phone devices belongs to the group of known people or community or only within a family.

1.5 Scope & Objectives

There are many cloud storage service providers in market. Each provider has different policies and implementation to handle cloud storage services. There are many challenges with existing cloud storage such as user has no idea about service provider created how many backup copies of data, only service provider has full control on data. There are various issues related to ownership of data, capability, efficiency, compliance and security. User is also not aware about actual location of physical storage. So there is scope to extend the existing Cloud Storage concept to build private Cloud Storage on which user has full control and aware about where and how its data is kept. Objective of this thesis is to remove the dependencies on Cloud Storage service providers and avail user a private cloud storage build using the unused storage space of mobile phone belong to the group of known people. This way user is aware about where and how its data managed.

Currently if user needs to share any file, he has to do it manually by transferring file via Bluetooth or Wi-Fi even though all users are connected to the same local network (WLAN Network). PMS-Sharing overcomes this problem of manual transferring and authenticating. In PMS-Sharing, we introduce a framework for automatically authenticating users in a group to allow sharing storage. We systematically used RSA encryption algorithm along with randomly generated pair key. Pair key is unique for each user, helps Group Owner device to identify and ensures that an authenticate user is accessing the group's data. Public

key cryptography based on Rivest–Shamir–Adleman (RSA), is used for encrypting messages to securely exchange pair key with designated user. It is using two level of encryption one at group level and another at user level. In PMS-Sharing, the user can share the directory among all other users in a group and other user can download and view the files of their choice easily by clicking on file name at displayed in list of files. User can do multiple functions like add, delete and download view or play files. PMS-Sharing will introduce a new framework to utilize the unused mobile phones as storage nodes to build big storage space which can be deployed as a shared storage on distributed network.

1.6 Thesis Outline

In chapter-1, we covered introduction of cloud storage and its high-level architecture, background of motivation of this thesis, scope and objective of this thesis. In chapter-2, we discussed in detail about existing cloud storage concept, its advantage and disadvantage, brief overview of important cloud storage applications and statement of problems which we have found in existing cloud storage sharing. In chapter-3, we discussed about proposed work, PMS-Sharing feature specification and Application architecture. In chapter-4, we evaluated the PMS-Sharing using test case studies, and in chapter-5, we discussed conclusion and future scope of this work.

Chapter 2 Literature Review

2.1 Cloud Storage System

Cloud storage means storing of data in a remote location that is accessible from any device. Cloud Storage will improve efficiency and productivity in terms of backing up and securing the data. It has many benefits and Business can pay only for storage they require [8].

In Cloud, storage system data is stored on multiple third-party servers, rather than on dedicated server used in traditional network data storage. Third Party service providers have entrusted with users' data, and for security purpose, the exact storage locations of these data are unknown to most people [7]. When storing data, the customer "sees" a virtual server, hence it appear that data is stored in a particular place with a specific name, but such a place does not exist in reality. It is just a pseudonym used to refer a virtual space carved out in the Cloud [7]. Therefore, user data can be stored anywhere in the world and stored data may differ from time to time as cloud provider manage their storage. There is a different kind of cloud storage like a private, public, hybrid, many cloud storage providers provide service to store user data, and they provide different pricing for these services. We will discuss about some of the Cloud Storage application in section 2.x

Cloud Storage is categorized based on how data is managed on servers-

1. File Storage,
2. Block Storage, and
3. Object storage.

There are many types of File Storage with different characteristics like structure, access method, speed, security, and size. File storage is applied on local as well as network storage devices. Some necessary network file storage is Network File System (NFS), Server Message Block(SMB)/ Common Internet File System (CIFS) and Plan 9 File system protocol (9P) [1], [8]. Block Storage provides a way to store data in evenly-sized memory blocks, used to host the database, which supports random read and write operations [2], [8]. Amazon Elastic Block Storage (EBS) is developed for use with Amazon Elastic Compute Cloud (EC2) to deploy a broad category of data loads such as databases, containerized applications, big data analytics engine, file system, media data, and enterprise application data [9].

Object Storage provides an easy way to store unstructured data in the form of Objects. An object is a simple repository with the unique ID number, metadata, and actual data [3]. Amazon Simple Storage Service (Amazon S3) is capable of managing a significant amount of unstructured data. Table (1) represents the basic differences between file storage, block storage, and object storage system.

Attribute	Storage System		
	<i>File</i>	<i>Block</i>	<i>Object</i>
Use Cases	File Sharing, Local Archiving, Data protection	Databases, Email Servers, RAID, Virtual Machines	Big Data, Web Applications, Backup Archives
Access	Full path should be known, Directory, sub directory, folder and file name	Provide Direct Access to the block	No direct access to object. An Object storage system places a file-system bridge [Gateways] in-between the two.
Metadata	Not metadata, full path should be known.	No additional information, provide direct access to block	expandable and variable amount of metadata
Ease of Use	Simple to use and setup	Block Storage provide ability to incrementally edit any part of the file	Object storage is powerful and customizable because variable amount of metadata of any type
Client Side Visibility	Visibility of files and folders is same on client and server	support formatting of file systems like NTFS or SMB, NFS, VMFS (VMware) as need	Expose object storage as Network File System via Storage Gateways.
Recommendation	Recommended for shared storage	Recommended for dedicated storage.	Recommended for unstructured data storage

Table 1: Differences between File, Block and Object Storage System

2.2 Review of Cloud Storage Providers

IDrive [10], pCloud [11], Mega [12], Microsoft OneDrive [13], iCloud [14], Google [15], Box [16], NextCloud [17], DropBox [18] and SpiderOak [19] are some of the cloud storage service provider which provide some GB's of storage free of cost. Table (2) represents the basic review of these popular Cloud Storage Applications. Cloud storage demand is increasing in proportion to the number of mobile device users are increasing each day. GrandStore [19] proposed a system where user can store the credential of all cloud storage account to expand the free storage capacity and a way to manage this all free storage account from system GrandStore application.

Providers	Review Parameters				
	Collaboration	Mobile Application Support	Free Storage Size	Benefits	Drawbacks
Dropbox	Yes	Yes	2 GB	<ul style="list-style-type: none"> Ease of use Simple Sharing Data Recover 	<ul style="list-style-type: none"> Lowest amount of free storage Email invitation is needed for file sharing
Google Drive	Yes	Yes	15 GB	<ul style="list-style-type: none"> Built-in document editor Allow commenting on files Allow backup email attachment Highest free storage space 	<ul style="list-style-type: none"> Must setup Google Drive web application for sharing files
Microsoft OneDrive	Yes	Yes	5 GB	<ul style="list-style-type: none"> Built-in document editor 	<ul style="list-style-type: none"> Less user friendly
Box	Yes	Yes	10 GB	<ul style="list-style-type: none"> User friendly Support file tagging Higher Security 	<ul style="list-style-type: none"> No file synchronization from Computer to Box. 256 MB File size upload limit in free plan.
IDrive	Yes	Yes	5 GB	<ul style="list-style-type: none"> Lockup with passcode Best for backup 	<ul style="list-style-type: none"> No built-in tool for file editing.
pCloud	Yes	Yes	10 GB	<ul style="list-style-type: none"> Inbuilt music player Fast upload and Download speed 	<ul style="list-style-type: none"> No built-in tool for editing files on Mobile
Mega	Yes	Yes	50 GB	<ul style="list-style-type: none"> Provide big free storage Open Source Inbuilt video player Mega Chat feature 	<ul style="list-style-type: none"> Not Stable Customer review are bad, some customer lost their data
iCloud	Yes	Yes	5 GB	<ul style="list-style-type: none"> Rich Feature Set Inbuilt editing tools 	<ul style="list-style-type: none">
NextCloud	Yes	Yes	Unlimited (depend	<ul style="list-style-type: none"> Rich Feature set Inbuilt editing 	<ul style="list-style-type: none"> Storage size depends of users capability

			son provider)	<ul style="list-style-type: none"> • Open Source • private cloud solution • Can host using own hardware or provider server 	of hosting storage
SpiderOak	Yes	Yes	No Free plan	<ul style="list-style-type: none"> • Unlimited device s. • Good for Backup • No Knowledge encryption 	<ul style="list-style-type: none"> • No inbuilt editing tools

Table 2: Review of popular Cloud Storage Applications

Chetan et al [20] proposed a framework using OpenStack [21] software to deploy storage as a service private cloud model by securing using credential and data using Advanced Encryption Standard (AES) algorithm [22].

Adishesu et. al [23] proposed a solution to build the personal cloud using unused computers at the node. Virtual Machine is setup on each unused computer node to build a datacentre-less, distributed cloud.

Alginahi et al [30] explained the various methods of authenticating to guarantee the trusted flow of data, they described the various challenges of authentication methods and usage of cryptographic algorithm in Cloud and IoT.

2.3 Advantages/Benefits of using Cloud Storage

Ease of use – Most of Cloud Services provide easy interface for sharing files. User can simply drag and drop their files in Cloud Storage folder and then files automatically get sync to the server. Most of Mobile Applications provide backup feature which can backup all mobile data automatically to cloud storage without user intervention on set intervals.

Simple Sharing – User can quickly send find to anyone by just sharing web link. Cloud Storage replaces the tradition way of file sending using email thus it saves upload time to the email server because send web link is just like sending text message.

Accessibility – Cloud Storage is always accessible on internet from anywhere.

Data Recovery – Cloud Storage provides alternative to backup important files. Individual user as well as Business Units can plan to backup their important files to recover it in case of lose of device or disaster situation.

Cost Efficient – Most of Cloud Storage services provide free storage up to some Giga Bytes and storage capacity can be extended by paying minimal amount. Greatest advantage in terms of cost saving is maintenance cost. On premise cloud storage require additional hardware setup, power backup and maintenance cost.

2.4 Disadvantages/Challenges of using Cloud Storage

Table (3) represents the disadvantages of cloud storage.

Key Challenges	Brief Description
Compliance and Security	<ul style="list-style-type: none"> • Threats of data leakage • Unique Credential • Security Threats during Data upload and download • No global policy to protect users cloud storage rights.
Ownership of data	<ul style="list-style-type: none"> • Only Cloud Storage service provideer has full control on data. • No idea about how many copies of data are created by service provider for backup and restore.
Capability	<ul style="list-style-type: none"> • Rehousing of data is slow • Complex process to change service provider.
Efficency	<ul style="list-style-type: none"> • Downtime of service provider • Dependency on Network Speed.
Private Cloud	<ul style="list-style-type: none"> • No pure private or personal cloud storage unless user has dedicated storage in his own premises which leads to high cost.

Table 3: Challenges of using Cloud Storage

2.5 Existing Cloud Concept

Figure (2) draws the high-level diagram to represent Cloud Storage on Internet Network. User devices connect as client devices on internet network. Cloud Service Provider setup and maintains Storage Gateway and Physical Storage in their respective business premises. User can access Cloud Storage using the provider client application. Cloud Storage provider also provide the SDK to enable developers to build their own client application using the API (application user interface) provider by service providers. Cloud Storage service provider provides the many API for various services such as signup, login, upload, download and push notification etc.

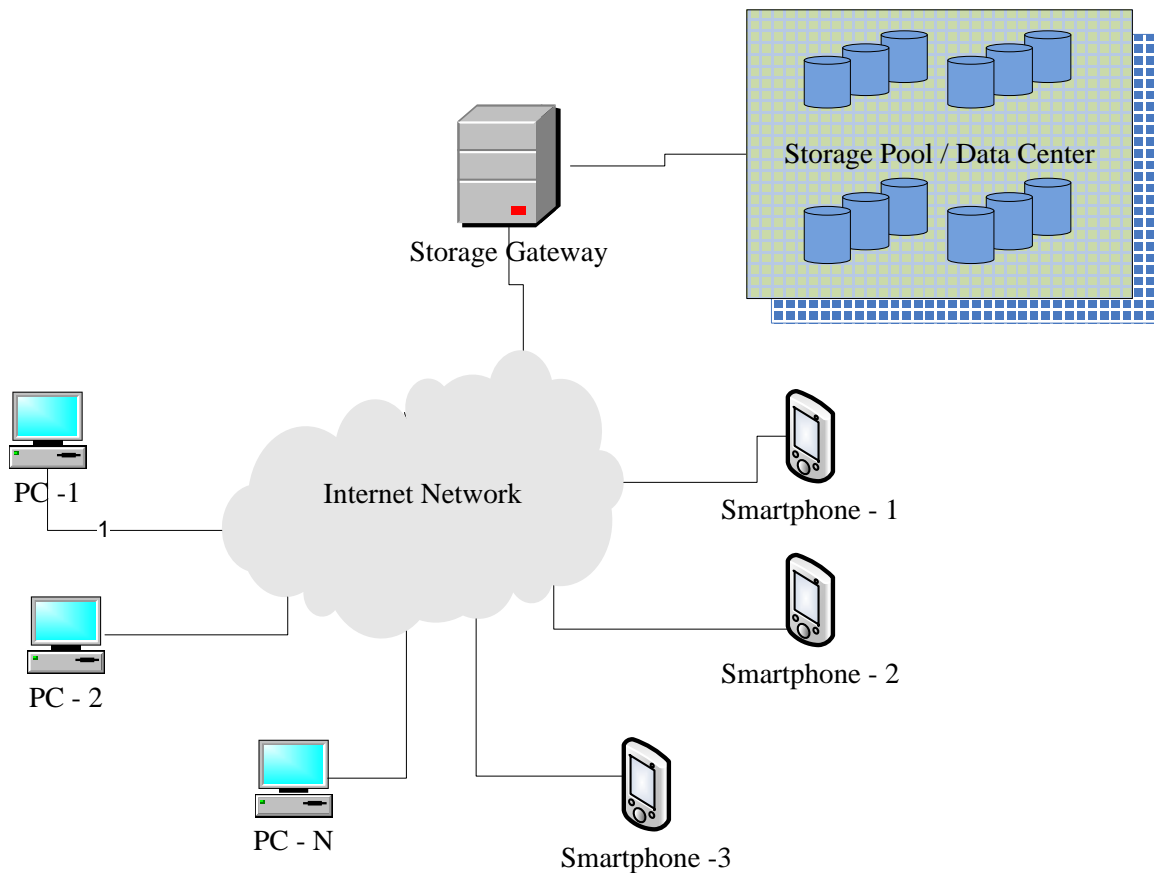


Figure 2: Existing Cloud Concept

2.6 Motivation & Problem Statements

We have studied the various aspect of existing cloud concept. We have identified few areas as listed below as a problem statement to work further to resolve them:-

- In existing cloud concept only provider has full control on user data. Users don't know physical location of the data, if provider fails, it is not easy to retrieve data from provider.
- In exiting file sharing methods, there is lot of manual process that user need to follow. For example in case of sharing file on Cloud ,user must need to login into the account thus require internet access all the time, then user need to manually choose the users to share the files and then send. Let discuss another example where user want to share files using Wi-Fi, then he need to first pair both device manually after choosing correct receiving device/user then only he can share the file.
- Cloud storage creates intermediate copies of user data about which user is not known.

- Server IP address is always static. It is mandatory to keep the server IP static otherwise client device may not be able to connect it.

2.7 Current Challenges and Limitations

2.7.1 Battery Consumption

Smartphones are still not matured enough in terms of current consumption, and obviously the smartphones which are operating in server mode need more processing and battery power to serve the request from n-number of users. There is threat of connectivity break with PMC-Storage when device power down.

2.7.2 Data recovery in case of loss of device

Even though we can use existing backup methods, there is need to derive secure and resource efficient method for Data recovery.

2.7.3 Connectivity

Even though 4G and 5G networks mitigate the connectivity issue still there are certain rural, hill stations and desert area where network is not deployed fully.

Chapter 3 Proposed Work

3.1 Overall Description of PMS-Sharing

The proposed system is called Personal Mobile Storage Sharing (PMS-Sharing). The system removes dependencies on Cloud Storage Service Provider to store the data on servers in order to access it anywhere around the globe. In proposed solution data are stored on the devices belongs to the known group of members, by proposed way user's data are safe and reachable physically in case network is not available due to some natural hazard.

In this era of smartphone devices, people are buying new smartphone in very short time and their old smartphones become useless. There is no proper way to recycle or utilize this useless smartphones. We proposed a solution to utilize these useless smartphone and their storage capacity as cloud storage. Network bandwidth and throughput has increased far better with the deployment of 4G and 5G networks, 4G provides data speed up to 100 Mega bit per second and 5G provides up to 10 Giga bits per second theoretical speed. On the other hand Wi-Fi Alliance [24] also introduced WiFi-6, next generation of highly efficient Wi-Fi network with theoretical speed up to 11 Giga bits per second; these all makes proposed solution feasible to deploy on private as well as public network.

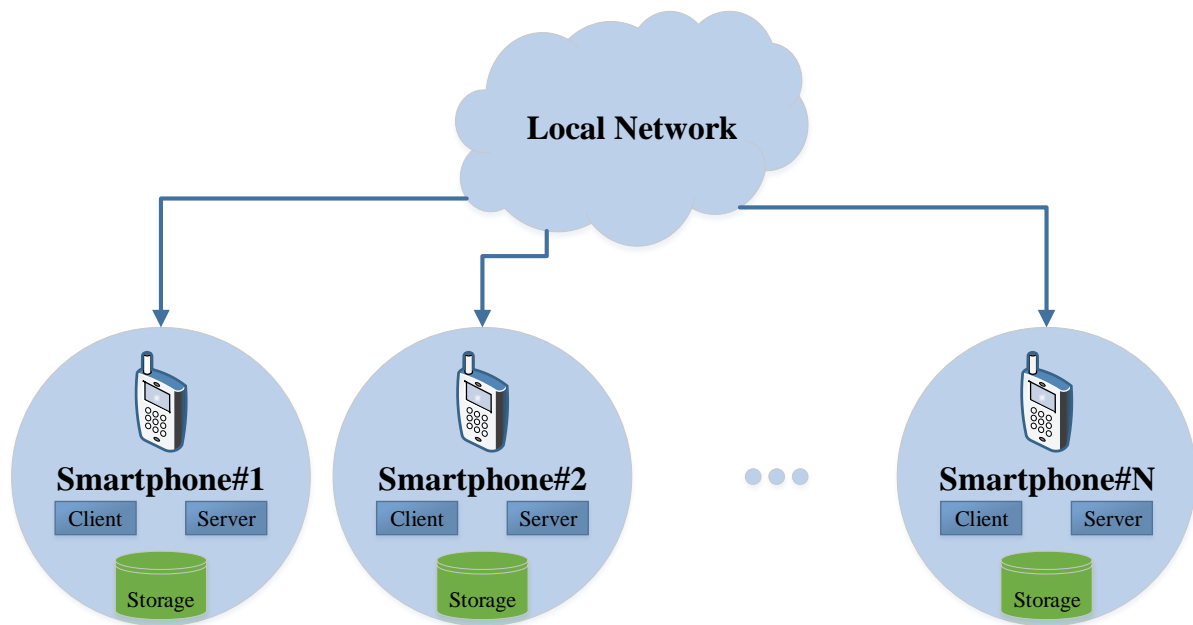


Figure 3: PMS-Sharing Concept

Unlike traditional cloud storage Figure (2), in proposed solution smartphones can participate as both way peers and hosting device as storage provider. Figure (3) draws the high level representation of the proposed framework. Smartphone users are creating and sharing lot of data using the advance features and services provided to them for example camera, audio stream recording, and data shared via messaging application and social media and there are duplicate files present among different users.

We are living in the era of smartphones, every user is having a smart phone with different storage capabilities such as 8, 16, 32, 64 and 128 GB of space and even more storage is going to be embedded in near future. We always find some user, which are not using even 50% of total available storage in their smartphone, which remains unused throughout the lifetime of Mobile device. For example if there are five members in the family and each member is having a mobile phone (i.e. device A, B, C, D and E). Suppose in devices respectively 40, 50, 80, 10, 120 GB of free space available. Therefore when all devices are connected as PMS-Sharing each user is having access to total 300 GB (40+50+80+10+120) of storage in that particular group of five people.

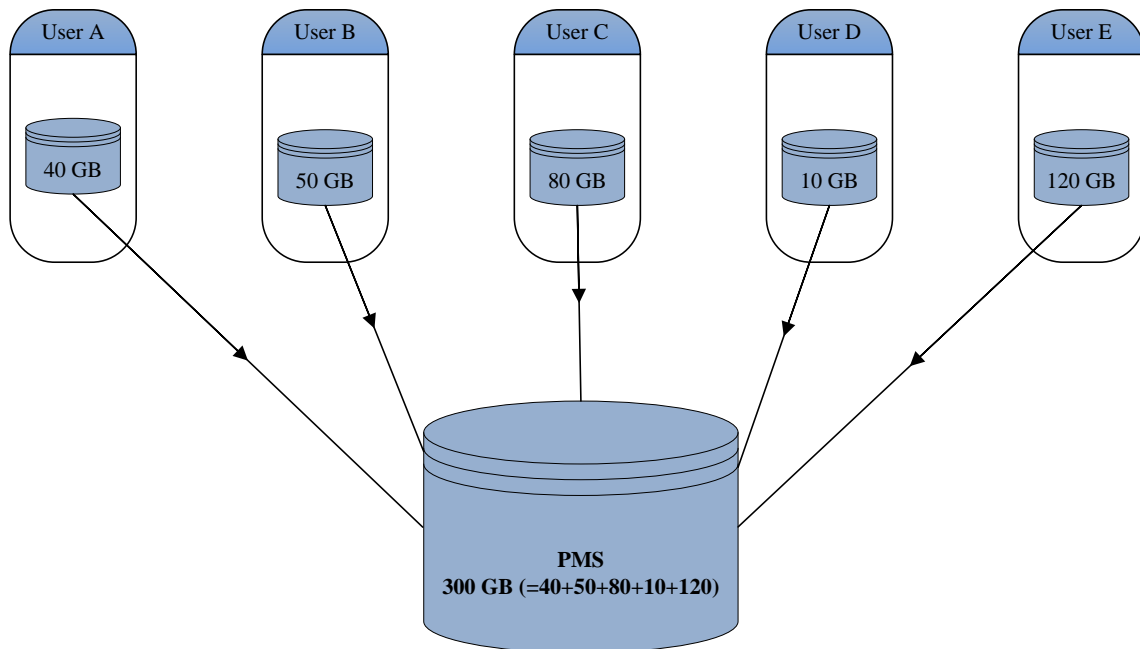


Figure 4: Aggregated PMS-Sharing Size

3.2 Perspective of PMS-Sharing

Providing a shared space for a group of user to share files (document, images, video etc.) among each other while connected to same network. Personal Mobile Storage Sharing (PMS-Sharing) enables user to form a local group of users and allow them to access shared storage directory.

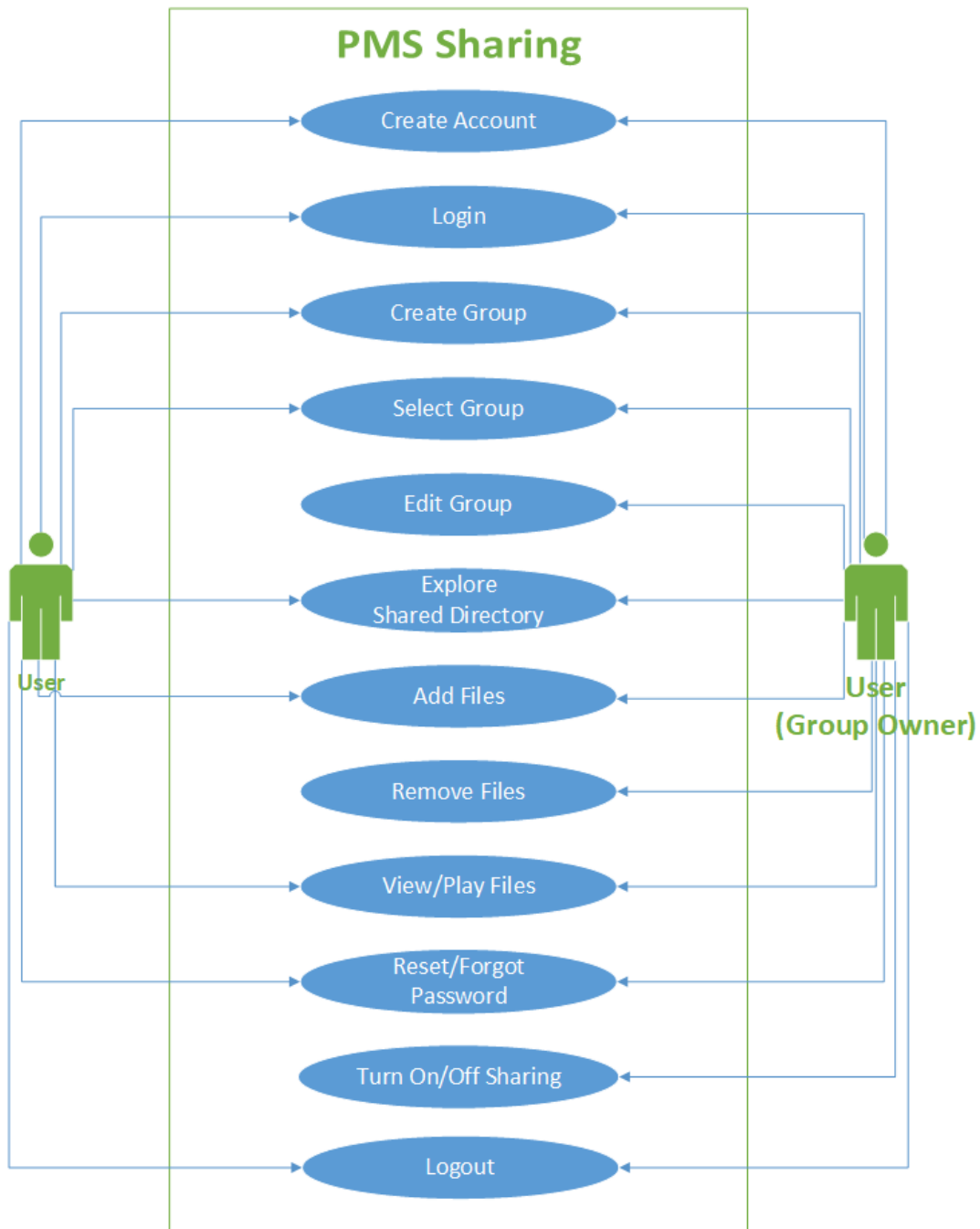


Figure 5: PMS-Sharing Use case diagram

Currently if user need to share any file, he has to do it manually by transferring file via Bluetooth or Wi-Fi even though all users are connected to same Wi-Fi Access Point (WLAN Network). PMS-Sharing overcomes this problem of manual transferring. In PMS-Sharing user can create a group and share the directory among all other users and other user can download and view the files of their choice easily by clicking on file name at displayed list of files. User can do multiple functions like add, remove and download view/play files.

3.3 Product Functions

PMS-Sharing will support following major function:-

1. User can create login account with valid email id.
2. User can login to the application with valid id (as created in 1).
3. User can create group of peer user whom with he wants to share the files.
4. Dashboard should list out all the Group created by user and Groups in which user is added.
5. User can modify the group details (Name, and participant user)
6. User should select the group to enter in shared directory of that group
7. User can add, remove and view the files in shared directory
8. User who creates the Group will be Group Owner by default for that particular Group.
9. The Group Owner can turn off and on sharing any time from Quick Menu Button.
10. User can logout from the application any time from any screen

Figure [5] represent the use case diagram for PMS-Sharing.

3.4 User Classes and Characteristics

Android Smart Phone users such as:-

1. Family Members - Family members can create a group to share photos, video and other documents.
2. Friend Circle - User can create a group to sharing files among friends.

3. Military and Secure Forces - Military and Secure Forces can use to share their confidential data within a define group of soldier.

3.5 Operating Environment

Android Operating System

3.6 Design and implementation constraints

All user's smartphone should be connected to same wireless network (using Wi-Fi Access Point or Mobile hotspot} to use main file sharing feature of this application. The maximum number of active users depends upon the maximum number of connections supported by the Wi-Fi Access point or Mobile Hotspot. For Example in case of Samsung Android Smartphone, Mobile Hotspot supports only ten simultaneous connection.

3.7 Feature Specification

3.7.1 User Interface

User interface will include following main screen through which user will interact with the PMS-Sharing application.

1. Login Screen
2. Sign Up Screen
3. Reset Password Screen
4. Dashboard Screen
5. Create New Group Screen
6. Edit Group Screen
7. File Explorer Screen
8. Button for turning ON and OFF sharing from Notification Panel

Login Screen is the first user interface from where user will start interacting with the application. Login Screen provide interface to Login and move to Dashboard Screen. Login Screen also has feature to reset password in case user forget password and feature to Sign Up as new user. Dashboard screen is the main screen where user can select the Group to open the shared folder created for that Group. Dashboard Screen also provides a Button to create new Group and option menu for each Group to modify Group details. Finally there will be a File

Explorer where user can see all files (Photo, Video, Document etc.). From Dashboard and File Explorer Screen user can logout any time. User will remain logged in till he manually sign out or user session is expired. Let's discuss about each screen in details in following section.

1. Login Screen : "Screen to Login into PMS-Sharing Application"

Figure (6) represents the login screen. Login screen contains Edit Text box to enter email id as User Login Id and password, Login screen will provide two clickable buttons "Login" Button and "New User" Button and two addition link button for "Forgot Password" and "Terms and Conditions". User can click on Login button to authenticate user to start application. Detailed constraints about Email Id and Password are explained in **section 3.7.1.2.**

Figure (8) depicts the Login flow. Note that password (B) is retrieved from database in encrypted form and Password (A) which is input by user it is also encrypted with the same encryption method as used at the time of user account creation. So at the time of account verification it just compares the encrypted digest of both password A and B. If encrypted password matches then screen will move to next Dashboard Screen.

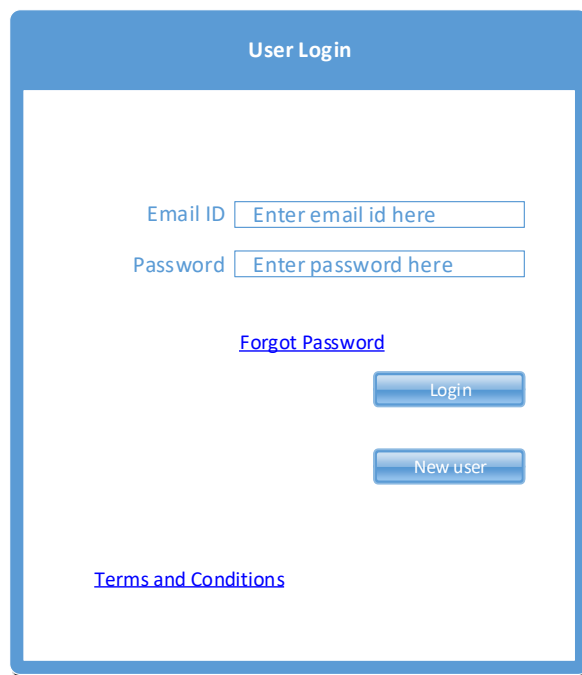


Figure 6: Login Screen

Figure 7: Sign Up Screen

2. Sign Up Screen : "Screen to create new user account into PMS-Sharing Application"

Figure (7) represents the screen to create new account. "Create Account" screen ask to input User Name, Email ID, Password, Re-type Password, dropdown list to select security question and input answer of security question. Screen will provide two buttons one for Create Account. Figure (10) represents the overall flow of signup screen. User can open the Sign up Screen by clicking on "New User" Button on Login Screen Figure (6). On Sign Up screen user need to enter the valid details and click on "Create Account" button. In case any entry in input fields are wrong or not meeting the input constraints then particular error message will be shown to the user. If user inputs the valid entries then user's details cross checked with existing database. If user is already a registered user then new account will be not created and a dialog message will be shown on the screen saying ***"User is already a registered user. Try to reset password using Forgot Password option or try again with other user details"***. If it is a new user then details are stored into the database, both password and answer of security question is being encrypted before storing into the database. There after "Sign up Screen" will be closed and user moved back to Login Screen. In PMS-Sharing application we are maintaining users database only on local mobile phone so it is mandatory

for user to remember the password or security question in case user forget password. If user forget both password and security question then there is no way to recover the account and user need to uninstall the application and install the application again to use it.

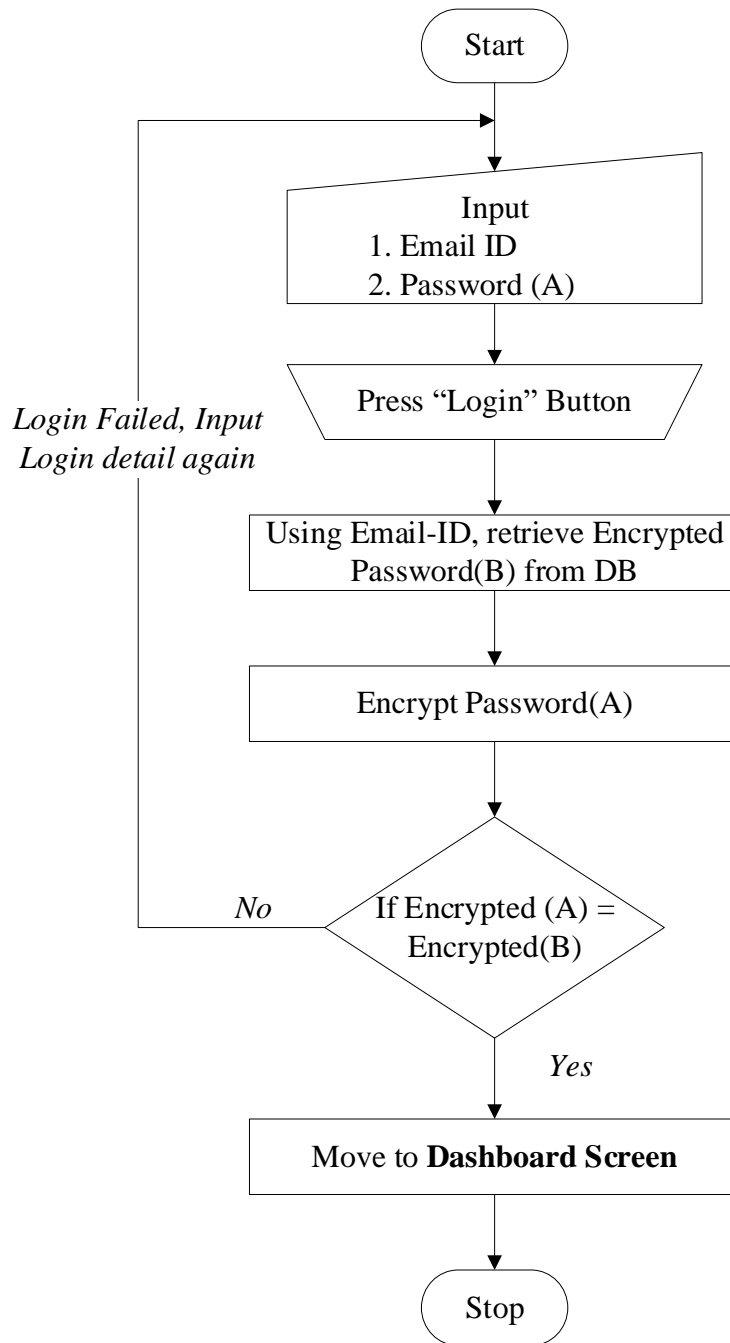


Figure 8: Login Flowchart

Characteristics of each field is defined in Table (4).

Field	Description and Characteristics
-------	---------------------------------

User Name	Name of the user, User name should not be more than 32 character, Special character are not allowed, only space character is allowed to separate two or three word used to represent the complete name of the user. Only space character is not allowed.
Email ID	Valid email id of the user, email id should not be more than 64 character
Password	Password should be alphanumerical with minimum 6 character length and maximum length can be 8 character, and at least one special character (@,#,_,%,^,&), Space or Tab character are not allowed in Password
Re-type Password	It should be same as Password
Security Question	Dropdown list of security questions for example :- 1. What is your nickname? 2. What is the name of first school joined? 3. What is your mother second name? 4. What did your father's occupation? 5. Who is your best teacher? 6. Who is maker of first Mobile phone purchased by you?
Answer Question	Answer of any one question selected by user, answer should not be less than 3 character and more than 32 character long.

Table 4: Input Constraint on User Signup Page

Reset Password

User Name

Email ID

Security Question ▼

Answer Question

Password

Re-type Password

Figure 9: Forgot password Screen

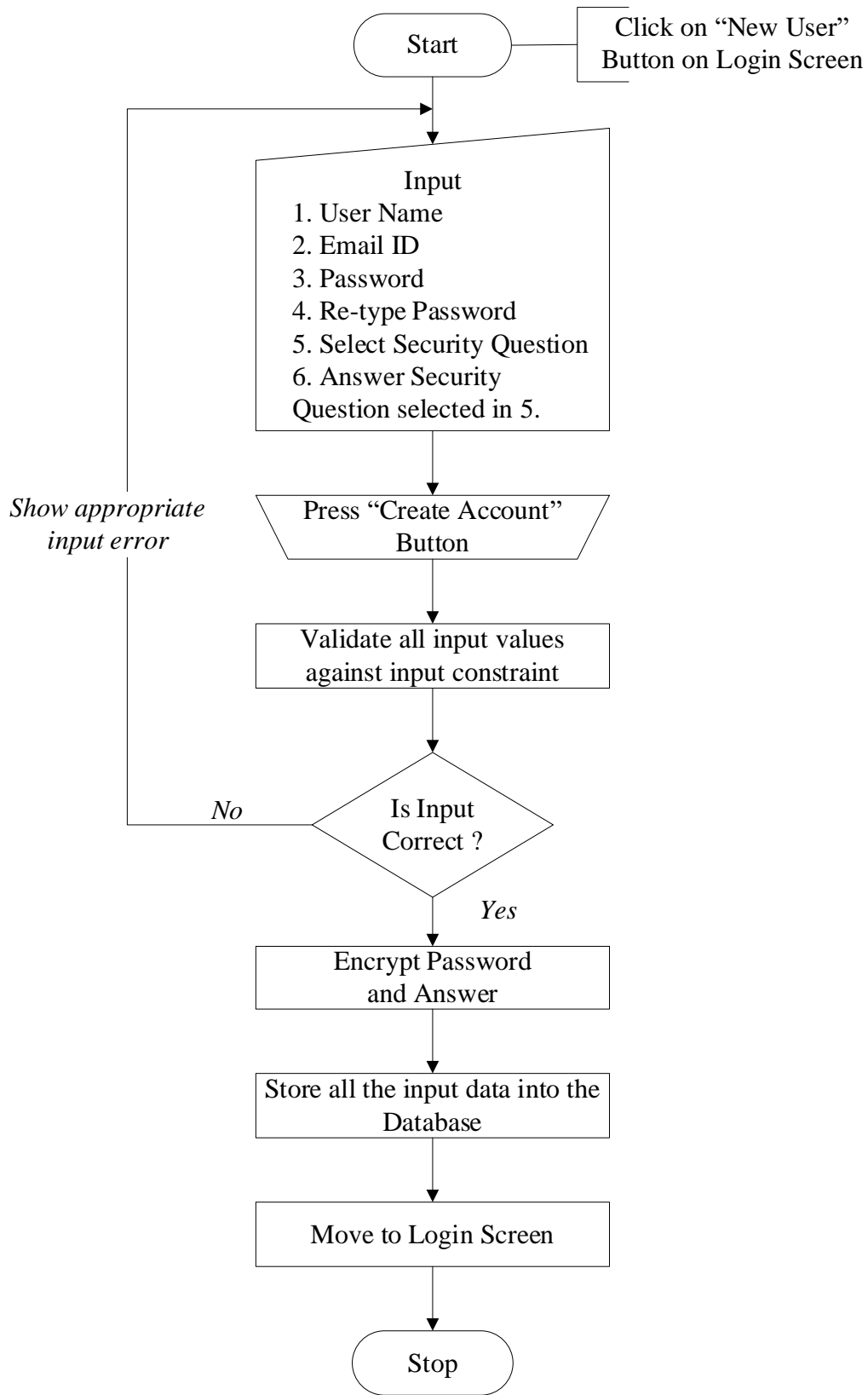


Figure 10: Sign-Up Process Flowchart

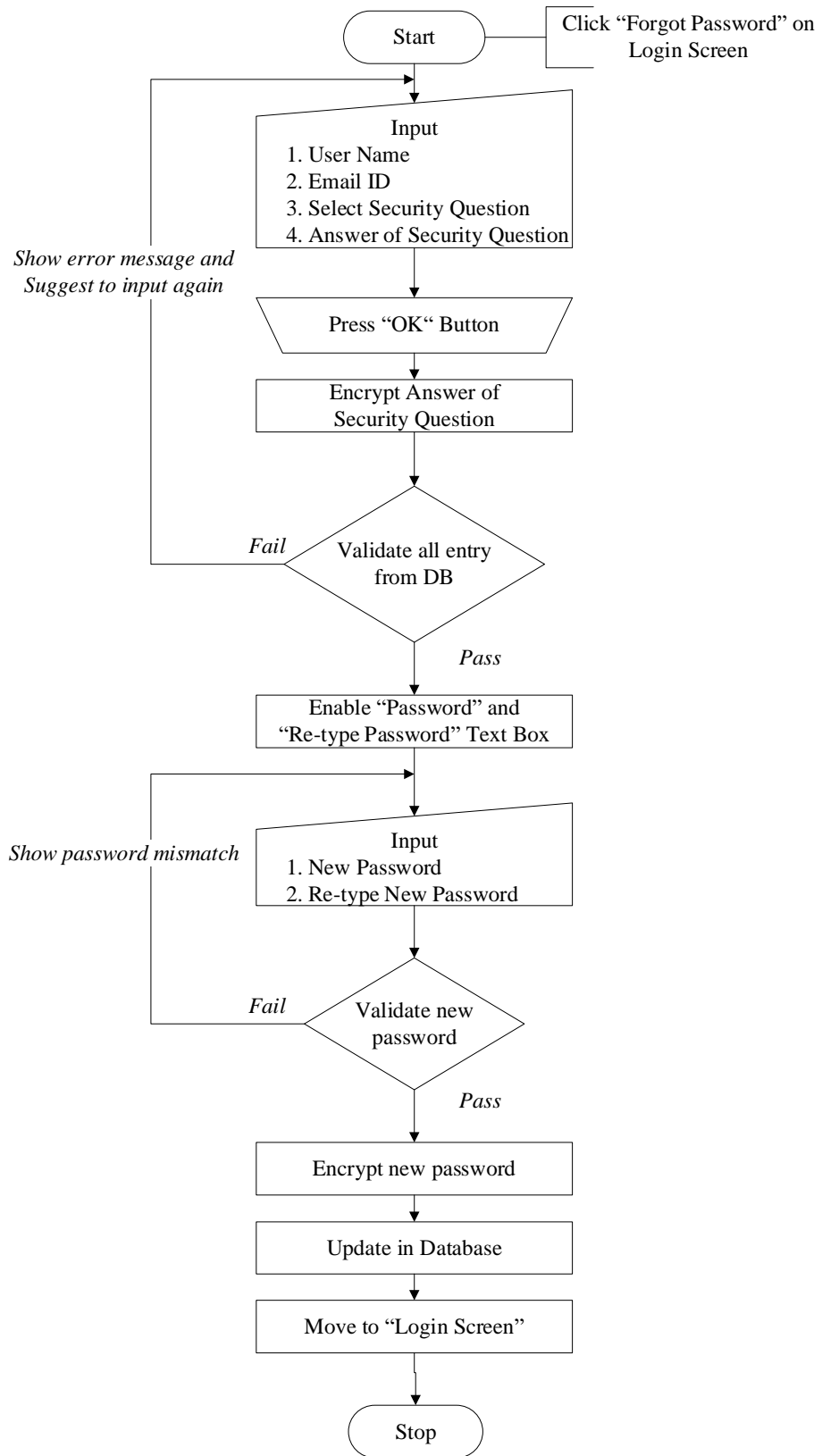


Figure 11: Reset password Flowchart

Figure (11) represents the complete flowchart for reset password mechanism. We are not storing actual password and security question answer anywhere without encrypting to keep the user data secure and at the time of verification we first encrypt the user input thus we can match the encrypted digest. Complete authentication process is explained in detail in section 3.8 System Design.

3. Reset Password Screen: "Screen to reset the password."

Reset Password screen provide a way to change/reset user password in case user need to either change the password or need to reset because he forgot the password. It must to remember security question and answer in order to reset the user password. Figure (9) represents the outline of reset password screen, where user can input User Name, Email ID, Select Security question and answer the same which user set during initial Sign Up process. After input detail user can press on OK button, if the input is valid user will be give option to type new password. After input password user can press update button to change his/her password.

4. Dashboard Screen : "Dashboard to display list of all the Groups"

Dashboard Screen displays list of all the Groups created by the user and list of Groups in which user is added by another users. Dashboard Screen acts as mediator. Figure (12) represents the wire frame of Dashboard Screen. Dashboard Screen provides following interfaces through which user can interact:-

1. Button to create new Group.
2. Option menu to Sign Out from the application
3. Option menu on each Group Icon to edit group details.
4. By default if user click on any Group, it opens the corresponding File Explorer Screen.

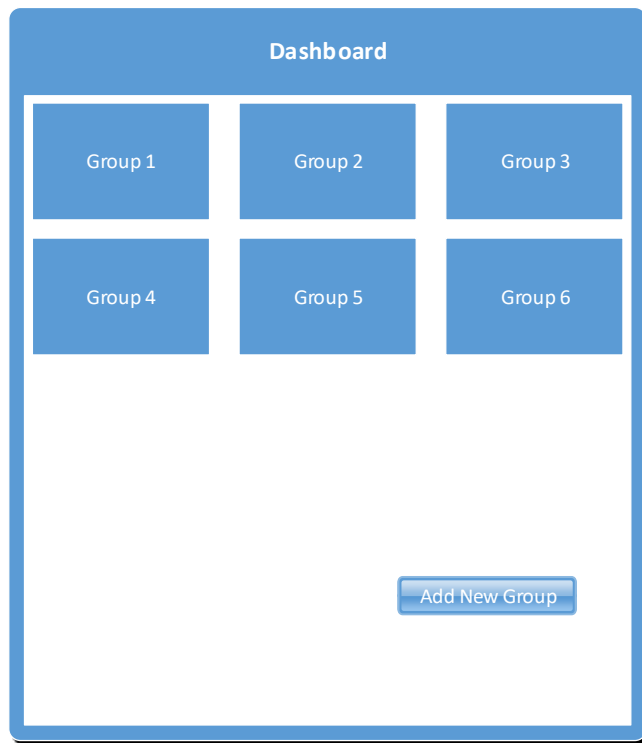


Figure 12: Dashboard Screen

Figure (13) represent one additional work that application do when user is on Dashboard Screen. On Dashboard Screen PMS Share application checks the state of control server and it starts the control server if server is in suspend or stopped. If the application is starting first time then it start the control server first. Control Server helps PMS Share application to establish control channel with other user in group, this control channel help to notify all the user in the Network whenever there is any new Group is created and formally it is starting point to start the any communication with any other user in the network. The detailed function of control server is explained in section 3.8 System Design.

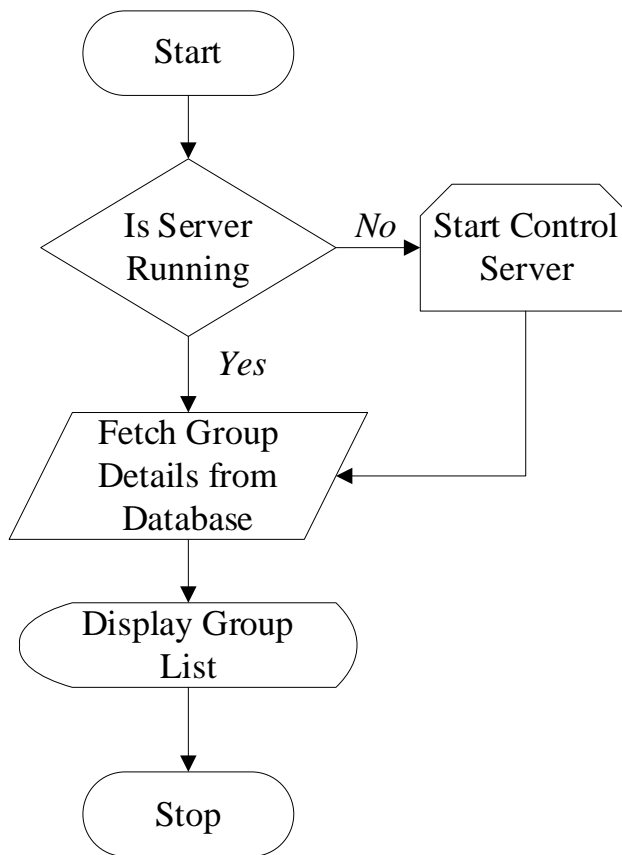


Figure 13: Dashboard Flowchart

5. Create New Group Screen : "Screen to create new Groups"

Figure (14) represent the screen where user can create a new group to share files. At Dashboard Screen user can click on "Add New Group" button to launch this activity where user will provided with the list of available users in the same network. There after user can select the user among the user list by clicked on checkbox shown in each row of particular user and input a unique Group name and click on "OK" button. Only those user can access the new group who are checked at the time of create group although user can add or remove any user later after creating this group using the "Edit Group" option on Dashboard Screen.

Figure (18) represents the basic flow of creating new group. The list of user are fetched from network using network service discovery and shown on the screen. Checked user list is picked by the system and stored in to the database for that specific group. After creating group and storing group details in database screen is switched back to the Dashboard Screen. User can also press on "Cancel" Button to come back to the Dashboard without creating any group.



The 'New Group' dialog box features a blue header with the title 'New Group'. Below the header is a section titled 'Select Users' containing a table with three rows: 'User 1', 'User 2', and 'User 3', each with a checked checkbox. Underneath is a 'Group Name' label followed by a text input field containing the placeholder text 'Enter group name here'. At the bottom are two buttons: 'Cancel' and 'OK'.

Select Users	
User 1	<input checked="" type="checkbox"/>
User 2	<input checked="" type="checkbox"/>
User 3	<input checked="" type="checkbox"/>

Group Name

Figure 14: New Group Screen



The 'Edit Group' dialog box features a blue header with the title 'Edit Group'. Below the header is a section titled 'Select or unselect user' containing a table with four rows: 'User 1', 'User 2', 'User 3', and 'User 4', each with a checked checkbox. Underneath is a 'Group Name' label followed by a text input field containing the text 'Group 1'. At the bottom are two buttons: 'Cancel' and 'Update'.

Select or unselect user	
User 1	<input checked="" type="checkbox"/>
User 2	<input checked="" type="checkbox"/>
User 3	<input checked="" type="checkbox"/>
User 4	<input checked="" type="checkbox"/>

Group Name

Figure 15: Edit Group Screen

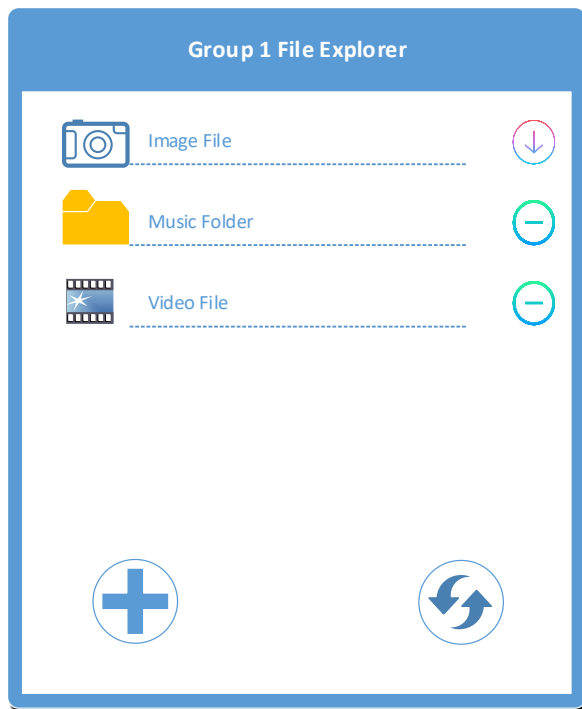


Figure 16: Local Directory View

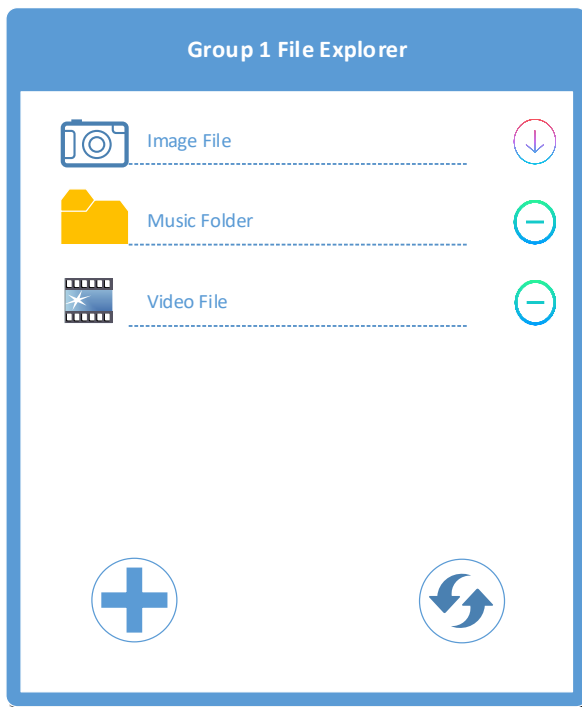


Figure 17: Remote Directory View

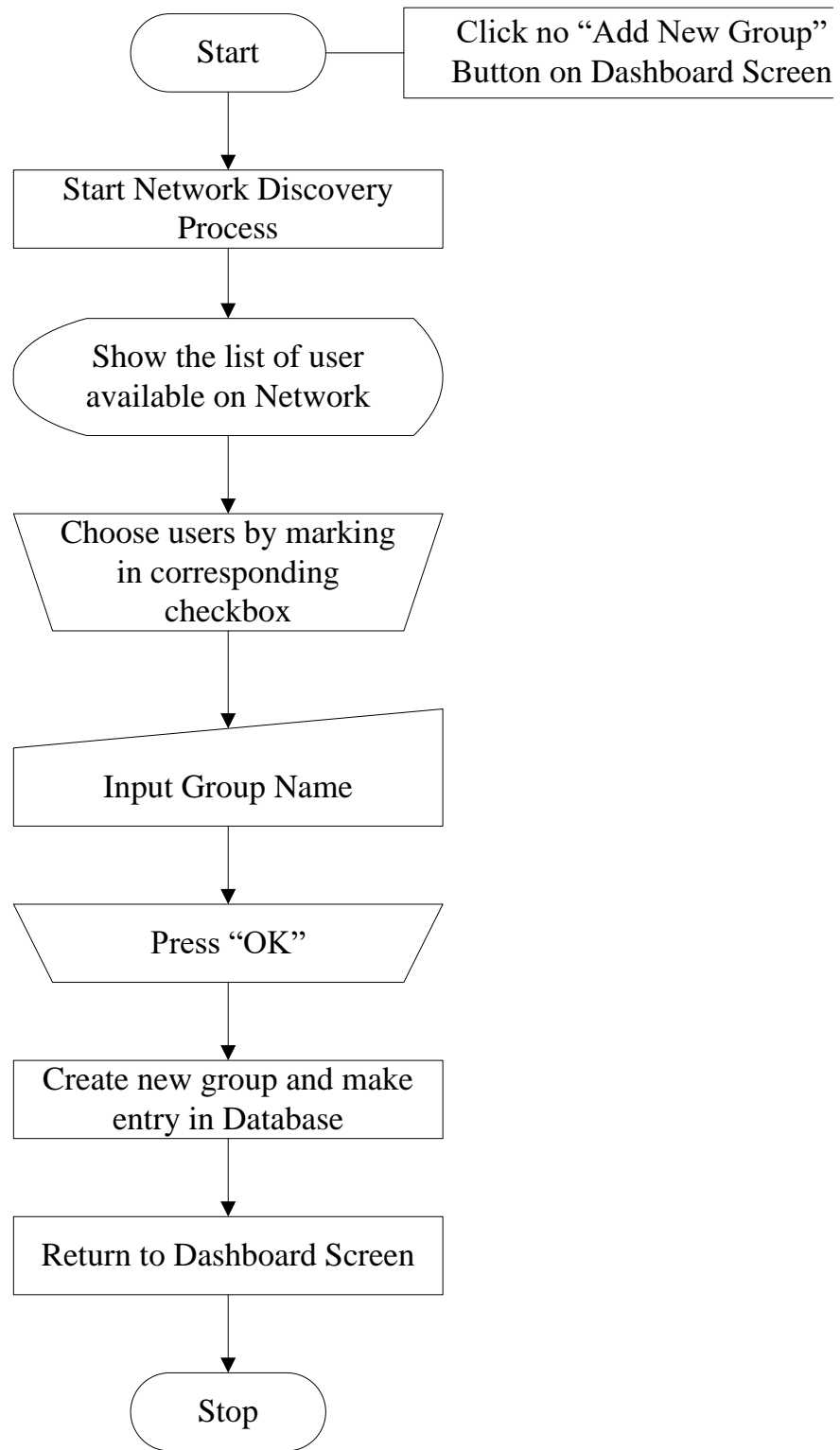


Figure 17: Flowchart to create new group

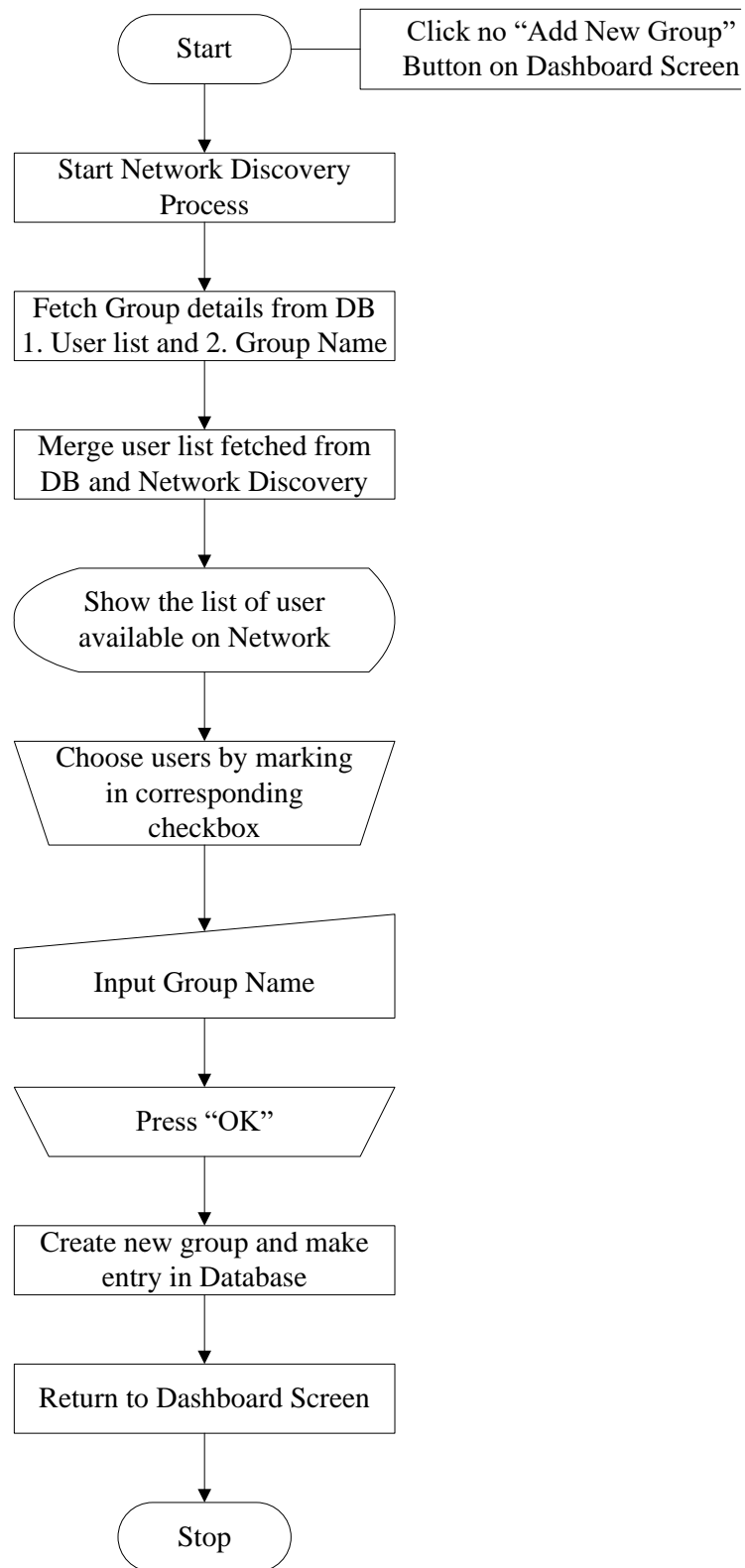


Figure 18: Edit Group Flowchart

6. Edit Group Screen : Screen to edit Group

Figure (15) represents the Edit Group Screen, where user can select and unselect the user and rename the group name. The process of Edit Group is similar to the mechanism of Create Group the only difference is that in case of Edit Group, the list of user is created by merging both the user list, one which are already present in the Group and another users, which are detected by network discovery process. Remaining process is same as depicted in Figure (19).

7. File Explorer Screen : "Screen to explore the shared directory of selected group"

File Explorer supports all types of file, and enables user to perform various actions like download, view/play. It provides two types of view screens:-

1. Local Directory View

This view will be opened when user is accessing the Local Group that is when user opens the Group which is created by him. It means user is the owner of that particular Group. As shown in Figure (16) File Explorer Screen display list of all files which are shared among all users added in that particular group. It also provides the button to add files and folder for sharing and it also provide button to remove file from shared directory. File explorer Screen will provide an additional refresh button to refresh the connection manually in case there is some problem in manual process.

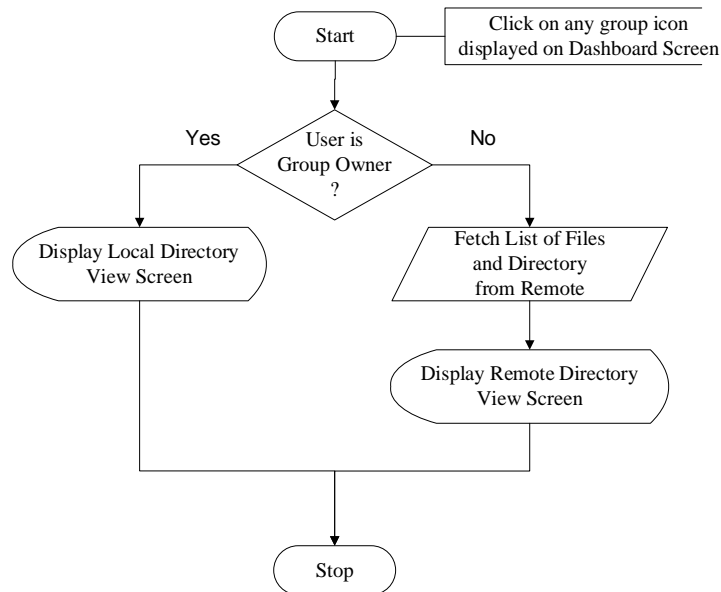


Figure 19: Remote Directory View

2. Remote Directory View

This view will be opened when user is accessing the Remote Group that is when user opens the Group which is created by some other user. It means user is not the owner of that particular Group. As shown in Figure (17) File Explorer display list of all files shared in this Group and it enables the download button for files which are not yet downloaded and remove button for the files which are already downloaded by the user.

9. Button for turning ON and OFF sharing from Notification Panel

Group Owner can enable and disable file sharing any time by toggling a button in Quick panel notification bar. When any user creates the Group by default File Server gets started. When user turns OFF sharing this file server will also get OFF and file sharing is no longer processed. Figure

3.7.2 Hardware Interface

PMS-Sharing application is developed only for Android [25] Smartphone Device. To user PMS-Sharing all the devices should be connected to Local Wireless Network. Wireless Network can be established using external routers (Wireless Access Point) or by turning on Mobile Hotspot in any of the user in Group and connect all other users of the group.

3.7.3 Software Interface

PMS-Sharing application runs on top of Android Operating System (version 9.0 and above). Thus Android API are broadly used in our application. Android is an operation system widely deployed on Smartphone which is highly portable and easy to operate. PMS-Sharing uses apache's FTP Server library (version 1.1.1) and Ftplet [26] library (version 1.1.1) supported and maintained by the Apache MINA Project [27]. FTP Server library is used to smoothly transfer files among multiple users within the controlled environment of PMS-Sharing framework.

3.7.4 Communication Interface

PMS-Sharing application uses FTP for Data Transfer Channel and HTTP for Control channel. Figure (21) depicts the communication interface between two PMS-Sharing application install on two different devices belong to different user. Command Interpreter is used to send and receive PMS-Sharing specific command and control messages.

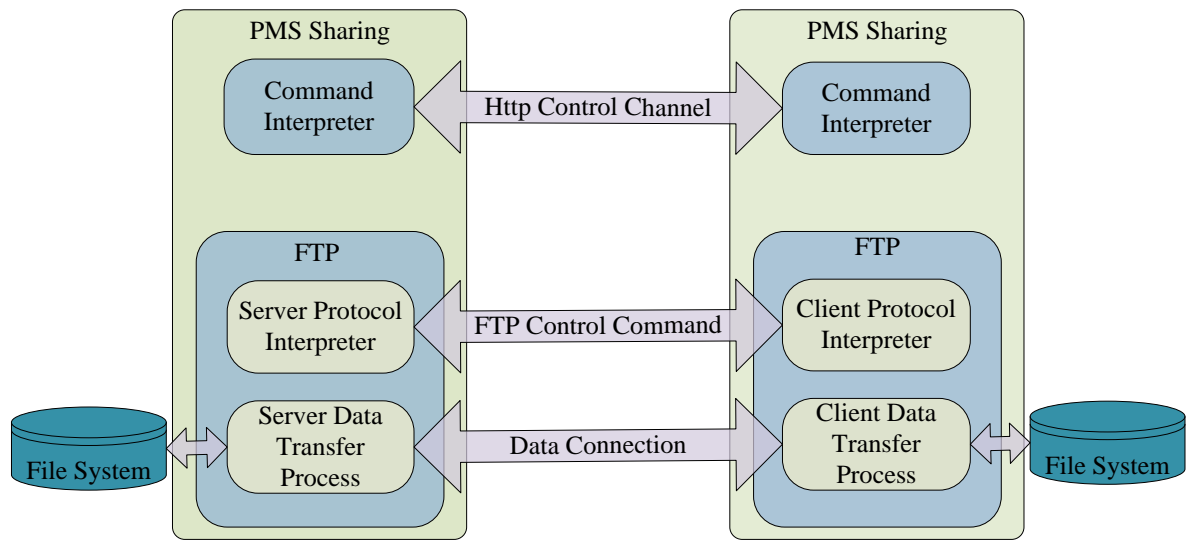


Figure 20: PMS-Sharing Communication Interface

Chapter 4 Implementation, Setup and Analysis

4.1 PMS-Sharing Architecture

Figure (22) represents the high level architecture diagram of the PMS-Sharing application. PMS Architecture has following main modules.

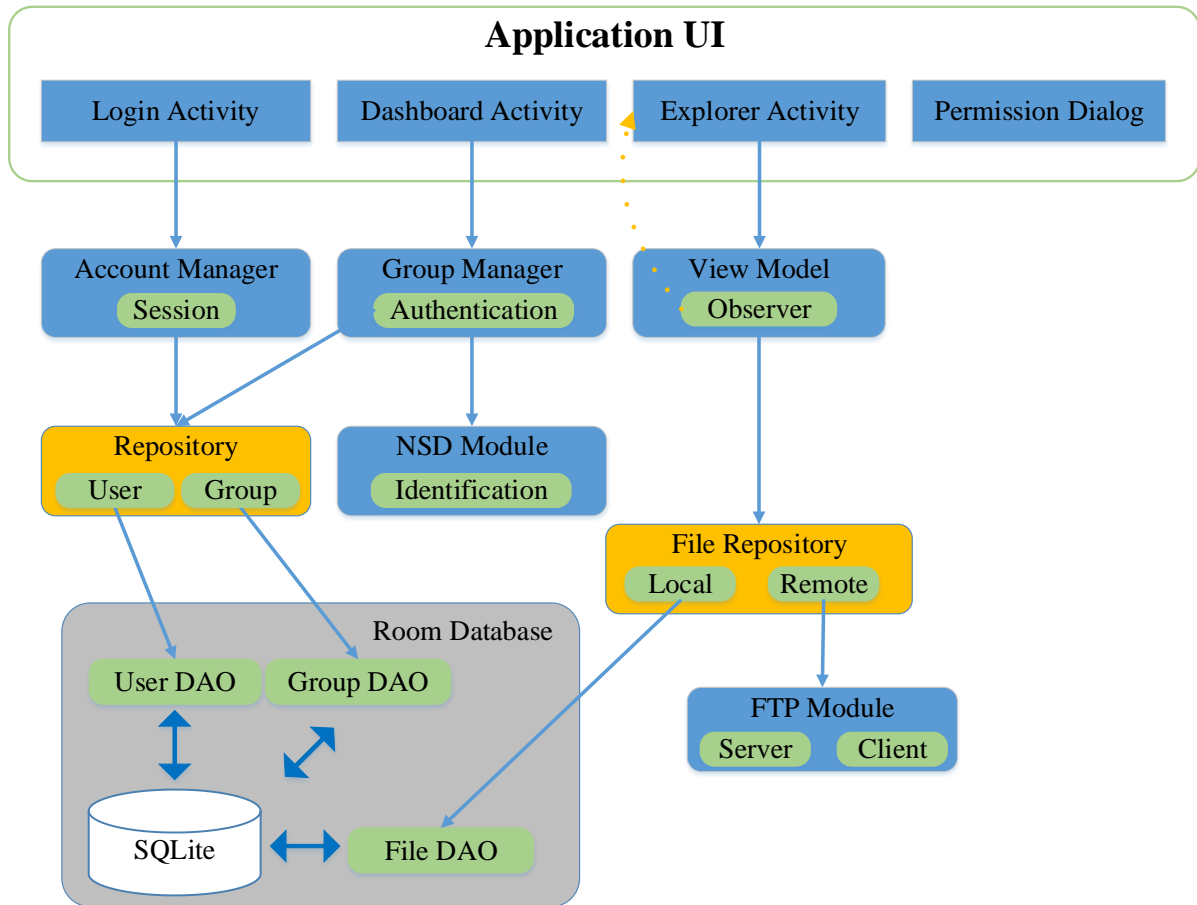


Figure 21: PMS-Sharing Architecture

- **Login Activity**

Login Activity control the user interaction with Login Screen, Sign up Screen and Reset Password Screen as explained in section 3.7.1 User interface. Login Activity is responsible for authenticating user to enter into the PMS-Sharing application and use it. Login Activity is dependent on **Account Manager**.

- **Dashboard Activity**

Dashboard Activity control the user interaction with Dashboard Screen, Create Group Screen, and Edit Group Screen as explained in section 3.7.1 User Interface. Dashboard Activity mainly dependent on **Group Manager**.

- **File Explorer Activity**

File Explorer Activity control the user interaction with File Explorer Screens (Remote and Local Directory View). File Explorer is responsible for providing rich experience to end user while using, sharing and accessing the files which are made available by the **File Repository**

- **Account Manager**

Account Manager provides services to the Login Activity to authenticate the user and granting Login Activity the necessary permissions to enter into the PMS-Sharing Application. Account Manager also manages the user session. Once the user is authenticated he is granted the access for 12 hours or till the user logout manually.

- **Group Manager**

Group Manager is at the core of PMS-Sharing. Group Manager provides all the services that enable all the function supported by the Dashboard Activity. Group Manager controls function of Discovering Client, Creating Group, Editing the Group and showing group details.

- **NSD Module**

Android provides the API to user Network Service Discovery. Using NSD PMS-Sharing application can easily identify and detect the users which are using PMS-Sharing services. NSD Module help Group Manager to populate all the users available on the same Network and using PMS-Sharing services. Figure (23) explain briefly the steps involved in registering and unregister services broadcasted in network. User device will start broadcasting PMS-Sharing service on local network after registering the PMS-Sharing services[A] and User device can also discover similar services broadcasted by other user on the same network[B]. On discovering any service NSD resolves the discovered services to

get IP and Port of the peer discovered device[C]. Then NSD can stop discovery any time by un-registering services [D].

- **FTP Module**

There are many way to send file from one device to other device using FTP (File Transfer Protocol), HTTP (Hypertext Transfer Protocol), FTPs (FTP over SSH), SCP (Secure Copy) etc. In our proposed method we used FTP for file transfer from one device to other device. When we want to send very large file over internet FTP comes first in mind. FTP can transfer single and can also transfer bulk files over TCP/IP. For running FTP protocol FTP client and FTP server required and FTP client can connect with FTP server using credentials like user ID and Password. Client ask for file and remote or local server provide it. FTP protocol that relies on two communications channels between client and server one is command channel and other is data channel. A command channel is used control for conversation using command and data channel for transfer files.

There are two type of FTP mode one is active mode and other is passive. In active mode, a client initiates a session via a command channel request, the server initiates a data connection back to the client and begins transferring data. In passive mode, the server instead uses the command channel to send the client the information it needs to open a data channel. Because passive mode has the client initiating all connections, it works well across firewalls and Network Address Translation (NAT) gateways. As explained in section 3.7.3 Software Interface, in proposed solution we used apache MINA libraries to use ftplet API for managing File Transfer using FTP protocol.

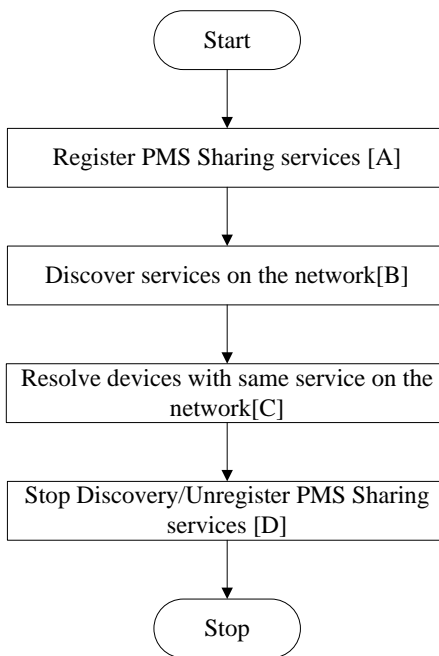


Figure 22: NSD Module

- **Database**

PMS-Sharing make use of Android Room API [28] for database management. In PMS-Sharing we need to manage the user Account details, Group Details and Remote File details. In Section 3.8.3 Database design for PMS-Sharing is explained in detail. For now giving a quick overview of User and File repository below:-

- **User Repository**

User Repository is used to read, write and modify user Account details and Group details.

- **File Repository**

File Repository is used to manage the remote file hierarchy and it also manages the list files added, downloaded and deleted/removed by the user.

4.2 Algorithm for Key Function and Features

4.2.1 Algorithm for Group Formation

Figure (23) represents the sequence chart for Group Formation. Group Formation is one of the complex algorithm in this application. As we discussed before any user can create a group, each group is identified with unique group id. Complete algorithm is explained in the below steps:-

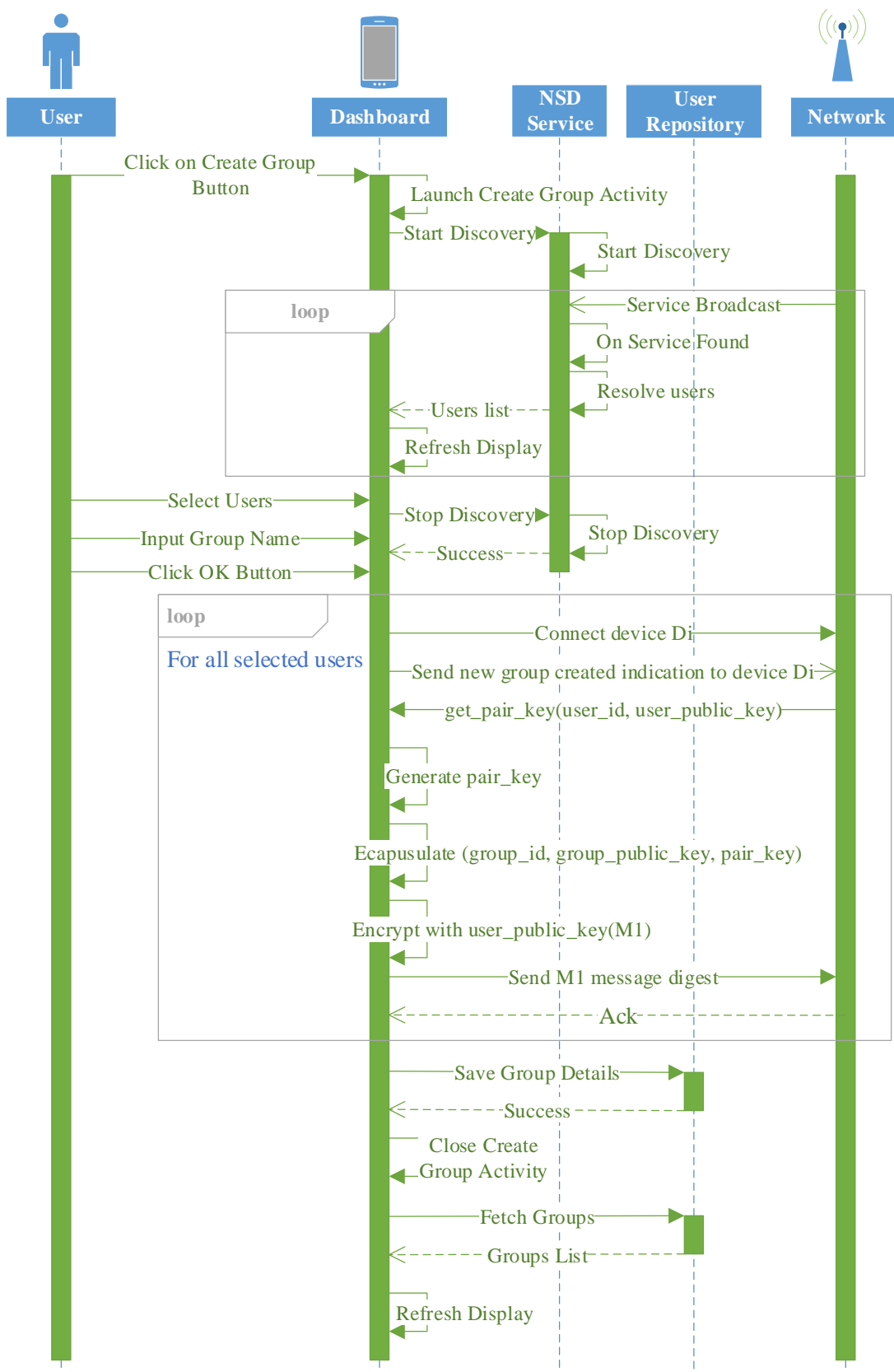


Figure 23: Sequence Chart Group Formation

Step 1:- Let there are 5 devices (d1, d2, d3, d4 and d5) connected to the local network.

Step 2:- Let there are 5 users (u1, u2, u3, u4 and u5) login into the PMS Service app with ids (uid1, uid2, uid3, uid4 and uid5) and currently accessing the Dashboard Screen of the app.

Step 3:- Let that the u1 wants to create a group to share files with other users.

Step 4:- Let that all devices already registered to NSD module and they are already broadcasting the service with name "pms_service"

Step 5:- user u1 starts group creation by clicking on create group button on Dashboard screen.

Step 6:- user u1 starts the NSD discovery.

Step 7:- all the devices (d2...d5) are discovered by the d1 and shown to screen.

Step 8:- Let user u1 selected user u3 and u4

Step 9:- Let user u1 gave group name as "group-1" and clicked on okay button

Step 10:- d1 will create a group-1 with id "group_id1" and stores selected users IP and PORT.

Step 11:- device d1 stops NSD discovery.

Step 12:- device d1 retrieve the IP and PORT of device d3

Step 13:- device d1 connects with device d3 and send notification of group availability

Step 14:- device d2 on receiving the notification, Requests d1 to share pair_key

Step 15:- device d1 receive pair_key request and stores user_id and user_public_key or requestor

Step 16:- device d1 generate the pair_key using group_id1 , user_id and random_number.

Step 17:- device d1 encapsulates the group information {group_id, group_name, pair_key, group_public_key} and encrypt with user_public_key and generates message digest (M1)

Step 18:- device d1 sends this M1 message digest to device d3.

Step 19:- on receiving M1, device d3 decrypt it using user_private_key and store information in database.

Step 20:- repeat **step 9 till step 17** for remaining user u3

Step 21:- group creation is successful and return back to Dashboard screen.

4.2.2 Algorithm to display Remote/Local Directory View

Figure (25) represents the sequence chart for Remote Directory View when user is access the remote group File Explorer display all the remote files. If somehow remote connection is not established in that case File Repository return Directory and File lists which user downloaded earlier that is files which are already present in local storage. Connect and Authentication Operation are covered in next section. Complete algorithm is explained in the below steps:-

Step 1:- Let user has already logged in PMS-Sharing application and currently he is on Dashboard Screen.

Step 2:- Let user selected a Group 'G1' on Dashboard Screen.

Step 3:- On touching at Group 'G1' icon, device starts process to fetch Directory View from File Repository.

Step 4:- File Repository first Connect and user Authentication is done with remote Group Owner. Complete Connection and Authentication process is explained in next section 4.2.3.

Step 5:- After successful authentication in section in step 4, device sends fetch request to Group Owner.

Step 6:- Group owner replies the Fetch request sent in step 5 by sending list of Files and Folder in the remote shared directory.

Step 7:- If step 6 fetch request fails, device requests fetch request to local storage to populate previously downloaded files.

Step 8:- Device populates the list of files and folder retrieved by either step 6 or step 7.

Step 9:- Now fetch is complete and user can perform file operations.

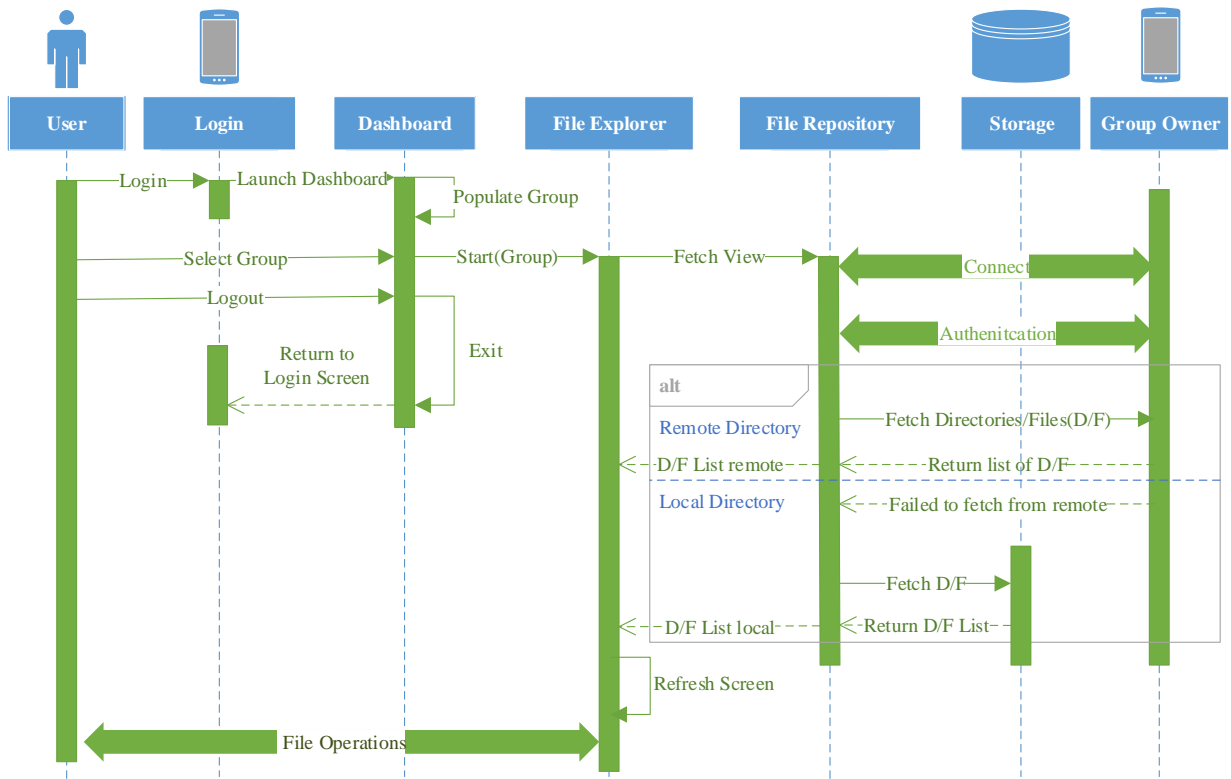


Figure 24: Sequence Chart Remote Directory View

4.2.3 Algorithm for Device Identification & User Authentication

In this section we will discuss about proposed authentication method, along with that we will also discuss the key factors that need to keep in mind before deciding about authentication method. Complete process is discussed in below section.

1. Device Registration

Each device registering PMS Service, will be visible by all other devices registering same PMS Service. When any device gets connected to the network, then all other user will automatically detect the device using NSD service discovery. In proposed solution default service name for registering in NSD Module is "pms-service"

```
private static final String DEFAULT_SERVICE_NAME = "pms-service";
```

2. Device Identification

Using "pms-service" name tag we are able to identify all the device in the network which have registering with same service name in NSD Module. Now next challenge is how to identify the role of the device like which device is operating in Group Owner mode and which device is just participating as user to achieve the same we further extended service register to identify server and client. The device registering with server role is Group Owner and all other devices are normal users.

```
public String mServiceName = DEFAULT_SERVICE_NAME;

public static enum TYPE {
    CLIENT,
    SERVER
}

public void registerService(String userName, int port, TYPE type) {
    ...
    if (type == TYPE.CLIENT)
        mServiceName = mServiceName + ".client";
    else if (type == TYPE.SERVER)
        mServiceName = mServiceName + ".server";
    ...
}
```

3. User Identification and Authentication

Now we know which all are the devices in the network registering PMS Service and who is the group owner and now next part is how to identify users belong to a particular group and if there are multiple group owner in the network how to identify the group owner in which particular user is added because user's IP and Port can change after each connect and disconnect from the Network. To address this problem we proposed an authentication protocol as shown in Figure (29). It consists of following steps:-

- At the time of group creation when Group Owner selects the users and click of create group, then Group Owner device starts user registration where he first connect

with user device and notify about new group. On receiving notification for new group, User device will send request to group owner for get pair key with his public key and user id. Generation of pair_key and message digest is shown on Figure (26).

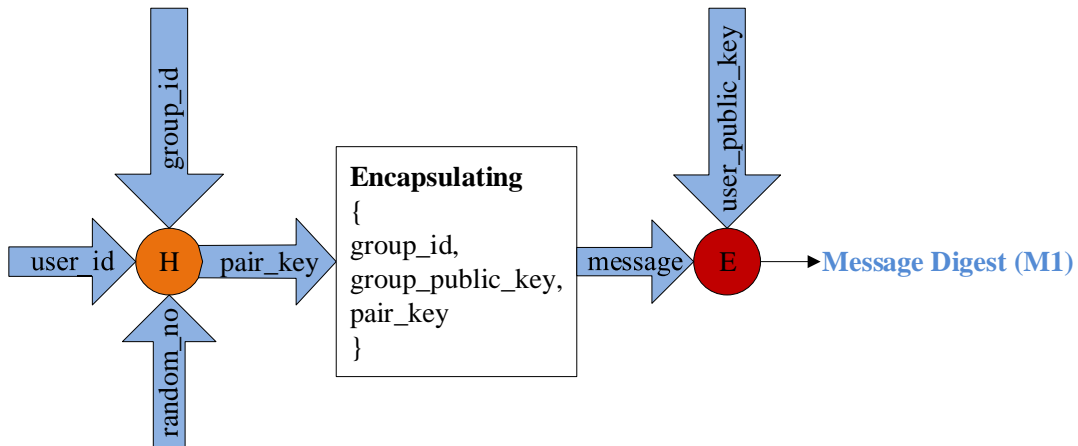


Figure 25: Generation of message digest on Group Owner Device

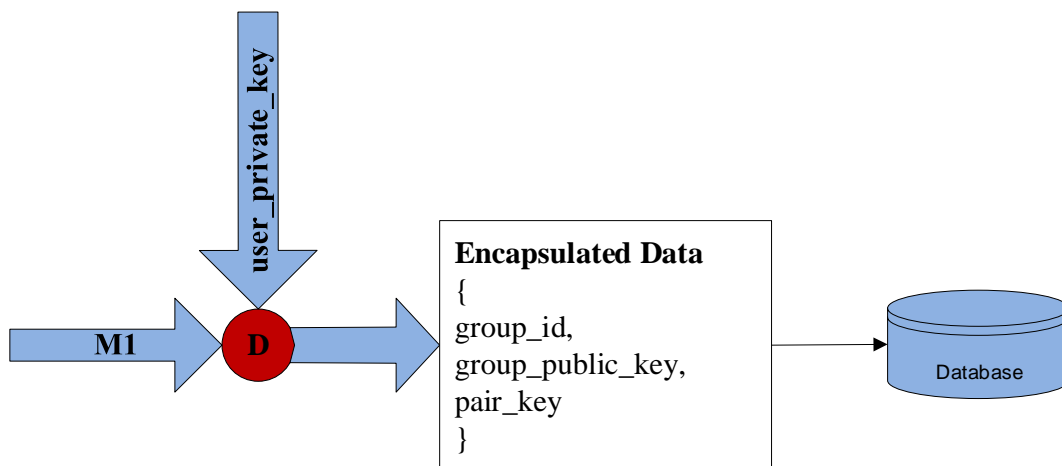


Figure 26: Decryption of message digest on User Device

- Now Group Owner device will generate a random number, and pass this user_id, group_id and random number to **Hash function (H)** to generate a 16 bit secured pair-key. Now it will encapsulate group_id , group_public_key and pair_key into single

message and encrypt the message using user public key to generate message digest (M1) and send to user.

- Then on receiving message digest Figure (27). User will decrypt the message digest using user private key to get encapsulated message {group id, group public key, pair key} and store this information in database.

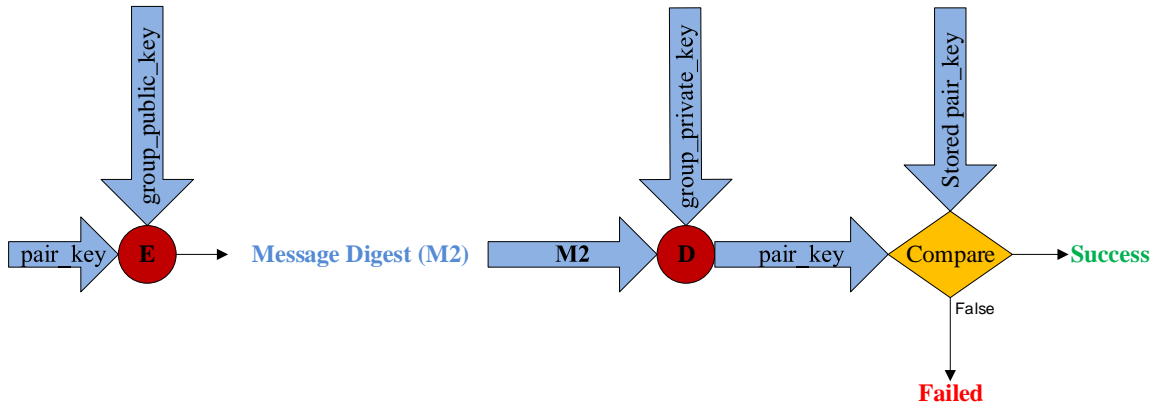


Figure 27: User Authentication

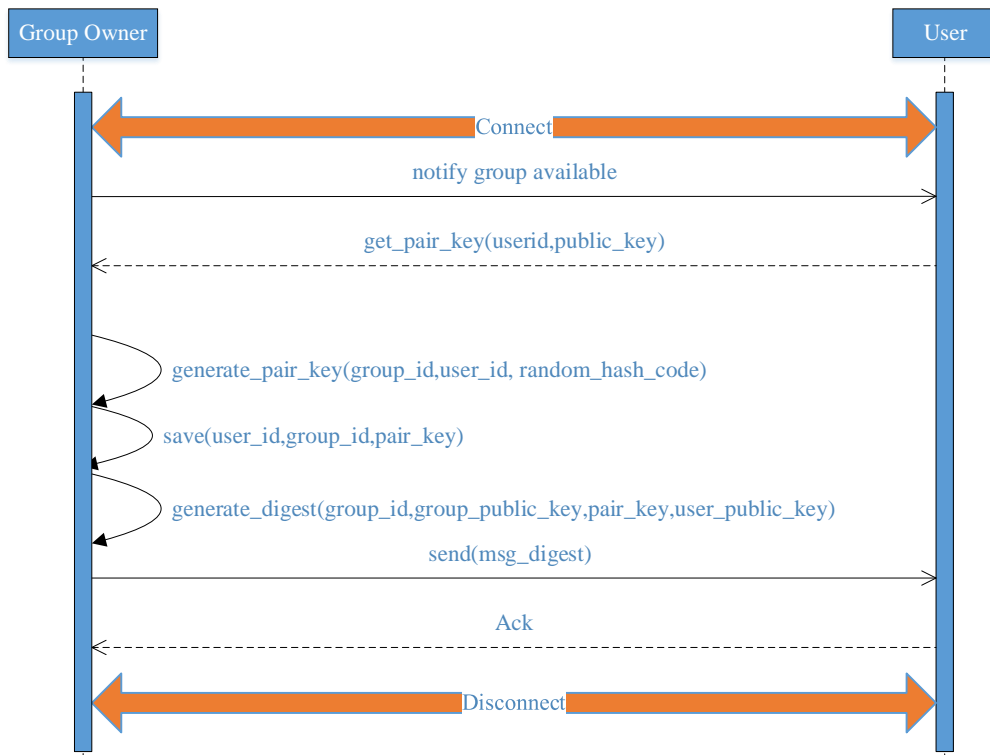


Figure 28: Sequence Chart Remote Directory View

- Now as shown in Figure (28) whenever user wants to access Group. User device will encrypt this pair_key using previously shared group's public key to generate message digest (M2) and send this to Group Owner. Group Owner decrypt this message (M2) using group's private key to get the pair_key and compare this pair key with the stored pair_key, if both are same means user is authenticated successfully.

4.3 Database Implementation

4.3.1 User Repository

We created User repository to manage the group management and user management. We incorporated three entity tables:-

1. Entity Name - "NetworkGroupDetails"

In this entity each record will store detail about group id, group name , total number of users in the group and user id of master user.

id(primary_key)	group_id(long)	group_name(String)	total_user(int)	master_user(string)
-----------------	----------------	--------------------	-----------------	---------------------

2. Entity Name - "NetworkGroupUserDetails"

In this entity each record will store information about each user of the group and it's associated pair key to authenticate the user while access the group.

id(primary_key)	group_id(long)	user_id(string)	ip(string)	port(int)	pair_key(long)
-----------------	----------------	-----------------	------------	-----------	----------------

3. Entity Name - "User"

In this entity each record will store information about user of the application.

id(primary_key)	user_name(string)	email(string)	password(string)	sec_q(string)	sec_a(string)
-----------------	-------------------	---------------	------------------	---------------	---------------

Where sec_q is security question and sec_a is security answer. Password, Security Question and Security Answer are encrypted before storing in database. We used apaches salted password encryptor which follow the principle explained here[29].

4.3.2 File Repository

File repository provides a protection wall between actual storage media and the shared folder thus the remote user is capable to access the files and folder added into the shared storage directory path. File repository also keeps track of file added by any remote user.

id(primary_key)	user_id(string)	group_id(long)	file_name(String)	file_path(String)
-----------------	-----------------	----------------	-------------------	-------------------

Here user_id is the id of user who added the file/directory in the group folder. group_id is the id of the group in which this file is added. file_name is the name of file or directory. file_path is the complete local path of the directory in which file is kept. In case it is directory it is the complete local path of the directory in which this directory is kept.

4.4 Setup, Evaluation and Analysis

4.4.1 Prerequisite

Following prerequisite are assumed to be exist for using PMS-Sharing application:-

- User should be able to use Android smartphone.
- All Smartphones should be capable of Wi-Fi Support.
- All Smartphones should be connected to the same Wi-Fi Access point or Mobile Hotspot to form a local network.

4.4.2 Setup PMS-Sharing application

PMS-Sharing application initial setup process is explained in below steps:-

Step 1:- Install PMS-Sharing application package.

Step 2:- Sign Up by providing user details and application password credential.

Step 3:- Login inside the application using credential set in Step 2.

Step 4:- Click on new Group.

Step 5:- Select users for the Group and click on Create Group button.

Step 6:- Start registering each user one by one and share group information with all users participating in group.

Step 7:- User can select the Group to explore and add/delete new files in shared directory.

Step 8:- Other user can also select the Group on Dashboard Screen to explore and add/delete new files.

Step 9:- Other user can click on any files to open and can click on download icon to download the files.

Step 10:- User can logout any time from any Screen using logout button in option menu.

4.4.3 Evaluation and Analysis based on case studies

Case Study 1:- Lets take an example of a family with five members Lajjesh, Nitesh, Gunjan, Bhumika and Urav. Suppose Lajjesh wants to share some photo graphs with all other family members. Lajjesh will perform following steps to share files.

Steps performed in normal file sharing:-

Step 1:- Lajjesh will choose files

Step 2:- Lajjesh will share via Wi-Fi Direct/Bluetooth and select user Nitesh, Device pairing request will be sent to Nitesh device

Step 3:- Nitesh will accept device pairing request and both devices are paired.

Step 4:- After pairing, Lajjesh Devices start transferring files to Nitesh Device.

Step 5:- Repeat Step 1,2,3 and 4 for Gunjan, Bhumika and Urav also.

Steps performed in PMS-Sharing:-

Step 1:- Lajjesh will login in PMS-Sharing application

Step 2:- Select Family Group

Step 3:- Add files which he wants to share with Nitesh, Gunjan, Bhumika and Urav.

Evaluation and Analysis:-

Evaluation Criteria	Normal File Sharing	PMS-Sharing
Number of steps	4 Members * 4 Steps for each = 16 Steps	Only 3 Steps
Authentication	Manual by pairing devices	Automatic once initial group is created.
Easy of use	Many steps	Simple steps

Medium of transfer	Wi-Fi P2P Ad-hoc network/ Bluetooth	Wi-Fi Infrastructure Network/ WLAN
Involvement	Both sender and receiver need to involve actively	Only sender is involved actively
Reliability	File transfer will fail if receiving device don't have enough space	Receiving device can view files without downloading if storage is not available for download.

Table 5: Evaluation using Case Study#1

So we can conclude that PMS-Sharing is better than using normal file sharing approach. PMS-Sharing helps in saving overall time and user involvement for sharing. In PMS-Sharing only sender need to add files in shared folder and other user can sync/download the file whenever they want to download unlikely to the normal file transfer where file is transferred immediately even though another user don't want to do so.

Case Study 2:- Lets take an example of an army base camp with seven soldiers- Lajjesh, Nitesh, Rohit, Mohit, Kapil, Harish and Anil. Suppose Mohit wants to share some secrete information files with all other soldiers. Mohit will perform following steps to share the files using cloud.

Steps performed to share files over Cloud Storage:-

Step 1:- Mohit will login to Cloud Storage application.

Step 2:- Upload files to Cloud Storage server.

Step 3:- Share link of uploaded files with other soldiers via some other medium either email or chat application.

Steps performed in PMS-Sharing:-

Step 1:- Mohit will login in PMS-Sharing application.

Step 2:- Mohit will selects the Group of Soldiers.

Step 3:- Add files in shared folder of the Group.

Evaluation and Analysis:-

Evaluation Criteria	Cloud Storage Sharing	PMS-Sharing
Number of steps	3 Steps	3 Steps
Authentication	Login using user credential	Login using user credential and Group level user identification and authorization with RSA encrypted packet.
Easy of use	Simple steps	Simple steps
Medium of transfer	Internet Cloud and URL of uploaded content shared using chat or email application	Wi-Fi Infrastructure Network/ WLAN
Involvement	Only sender is involved actively	Only sender is involved actively
Reliability	Receiving device can view files without downloading if storage is not available for download.	Receiving device can view files without downloading if storage is not available for download.
Security	Partially secured , any person who has account on cloud storage application and has the URL of shared files then he can easily access the files even though he is not the intended user	Fully Secured, Only member in the group can view and access the files.
User Control	Not control on data once uploaded to Cloud Storage, All the information is already exposed to the cloud service provider.	Full Control
Storage Location	Not known to the user	Known to the user

Table 6: Evaluation using Case Study#2

So in PMS-Sharing only 3 steps in total are needed, whole process of transfer and authentication is automatic once the initial group is created. PMS-Sharing helps in saving overall time and user involvement for sharing. In PMS-Sharing only sender need to add files in shared folder and other user can sync/download the file whenever they want to download unlikely to the normal file transfer where file is transferred immediately even though another user don't want to do so. Similarly if we compare with Cloud sharing where first user need internet access to use services, user can share uploaded contents URL to any other member, there is no control on data once uploaded to Cloud Storage, All the information is already exposed to the cloud service provider, any person who has account on cloud storage application and has the valid URL of shared files then he can easily access the files. While in case of PMS-Sharing user can use the services on local network without internet and the content is accessible only to the members of the group.

Chapter 5 Conclusion and Future Scope

Considering the current progress on Network connectivity and development of smartphones capable of high performance and equipped with Gigabytes of inbuilt memory. PMC-Storage can be considered as new horizon in Cloud Storage Space which can change the current dimension of cloud storage data center and expand it to every device exist on the planet with few Gigabytes of storage capability and network connectivity. PMS-Sharing removes dependencies on cloud-storage service provider to store the data on servers in order to access it anywhere around the globe. User's data is stored on the devices within the known group of family members, by this way his data is safe and reachable physically in case network is not available due to some natural hazard.

PMS-Sharing will be very useful application for the military, secure forces, corporate network, and for those users who do not want to upload their personal data with cloud-storage service provider because cloud storage can be unsafe and user does not know physical location of storage. PMC-Storage is very useful to deploy Private Cloud for a group of know people say a Community and for a Military Services.

5.1 Future Scope

Based on PMS-Sharing concept, personalized cloud storage can be built with marginal cost and more research ca be done to mitigate data recovery and security issues.

PMS-Sharing application can be extended in many ways; it can be made more useful for end users:-

1. Allow devices to store any file on any connected device.
2. Improve file access and store policy
3. Feasibility for Storage recovery in case any device get lost.
4. Smart algorithm to choose device to store new file automatically.

References

- [1] https://en.wikipedia.org/wiki/File_system, last accessed on: 15/09/2019.
- [2] <https://www.ibm.com/cloud/learn/block-storage>, last accessed on: 15/09/2019.
- [3] <https://www.ibm.com/cloud/learn/object-storage>, last accessed on: 15/09/2019.
- [4] <https://www.ibm.com/cloud/learn/file-storage>, last accessed on: 15/09/2019.
- [5] Michael Armbrust, Armando Fox, Rean Griffith, et al. "Above the Clouds: A Berkeley View of Cloud Computing" [R]. Berkeley, CA, USA: University of California, 2009.
- [6] Dai Yuanshun. "The Brief Review of Cloud Computing Technologies". Information and Communications Technologies. 2010.2, pp 29-35.
- [7] Isaac Odun-Ayo, Olasupo Ajayi, Boladele Akanle, Ravin Ahuja "An Overview of Data Storage in Cloud Computing". International Conference on Next Generation Computing and Information Systems (ICNGCIS).
- [8] Yuanyuan Guo, Jianjun Hao, Yijun Guo, Tao Luo, "Research on Data Block Storage Strategy in Cloud Storage System", Proceedings of 2017 3rd IEEE International Conference on Computer and Communications, 2017, pp. 2394-2397.
- [9] <https://aws.amazon.com/ebs>, last accessed on: 16-09-2019..
- [10] <https://www.idrive.com/>, last accessed on: 21-09-2019.
- [11] <https://www.pcloud.com/>, last accessed on: 21-09-2019.
- [12] <https://mega.nz/>, last accessed on: 21-09-2019.
- [13] <https://onedrive.live.com/about/en-gb/>, last accessed on: 21-09-2019.
- [14] <https://www.icloud.com/>, last accessed on: 21-09-2019.
- [15] <https://cloud.google.com/>, last accessed on: 21-09-2019.
- [16] <https://www.box.com/>, last accessed on: 21-09-2019.
- [17] <https://nextcloud.com/>, last accessed on: 21-09-2019.
- [18] <https://spideroak.com/>, last accessed on: 21-09-2019.
- [19] Li Zhang, Bing Tang, "GrandStore: Towards Large-Scale Free Personal Cloud Storage", Proceedings of 2017 International Conference on Computer Network, 2017, pp 118-123

- [20] Chetan Gaikwad, Bhoomika Churi, Kanad Patil, Tatwadarshi P. N., "Providing Storage as a Service on Cloud using OpenStack", Proceedings of 2017 International Conference on Innovations in Information Embedded and Communication Systems, 2017, pp 1-4.
- [21] <https://www.openstack.org/>, last accessed on: 24-09-2019.
- [22] B. Varsha and P. Suryateja, "Using Advanced Encryption Standard for Secure and Scalable Sharing of Personal Health Records in Cloud", International Journal of Computer Science and Information Technologies (IJCSIT), Vol. 5 (6), 2014.
- [23] Adishesu Hari, Ramesh Viswanathan, T. V. Lakshman, Y. J. Chang, "The personal cloud: design, architecture and matchmaking algorithms for resource management", Proceedings of the 2nd USENIX conference on Hot Topics in Management of Internet, Cloud, and Enterprise Networks and Services, 2012, pp 1-6.
- [24] <https://www.wi-fi.org>, last accessed on: 28-09-2019.
- [25] <https://www.android.com>, last accessed on: 26-01-2020.
- [26] <https://mina.apache.org/ftpserver-project/ftplet.html>, last accessed on: 26-01-2020.
- [27] <https://mina.apache.org>, last accessed on: 26-01-2020.
- [28] <https://developer.android.com/jetpack/androidx/releases/room>, last accessed on: 28-01-2020
- [29] <http://www.jasypt.org/howtoencryptuserpasswords.html>, last accessed on: 04-02-2020.
- [30] Alginahi, Yasser M., and Muhammad Nomani Kabir, eds. Authentication Technologies for Cloud Computing, IoT and Big Data. IET, 2019, chapter 3-4.