

CHAPTER 04

**A PEER-TO-PEER  
PROTOTYPE BASED  
ON RTMFP**

## 1. Introduction:

The design phase is perhaps the most delicate phase of the project since it will directly affect performance and the proper functioning of the developed tool which must be robust, efficient and reliable.

This chapter is devoted to the design phase of our model Peer-To-Peer. The goal is to develop a network for sharing files between users that guarantee minimum services offered by systems Peer-to-peer best known.

## 2. Operating environment of the project:

This system is created to give access to information shared by several students via a P2P network to offer students a way to exchange documentary by a reliable and effective based on the notion of reciprocity. The two main objective of this work are:

- Propose a way to organize students into group share called "peer group".
- Allow students to search, download and browse the documents they need.

To ensure the smooth running of this system of sharing, the application must consider the following points :

- Setting access permissions to files and folders by the elaboration of access structures to information and the management of machines resources.
- In this network the user does not need a centralized server to store and manage its data. In effect, through a specific software extremely easy to install, to configure and to exploit, which is semi-client, semi-server. Everyone can connect easily and efficiently.
- The routing to all data based on exchange agreements and reciprocity more or less formalized among the different actors.
- The user needs a method lies in the comfort and effectiveness research.

### 3. Exposition of the problem:

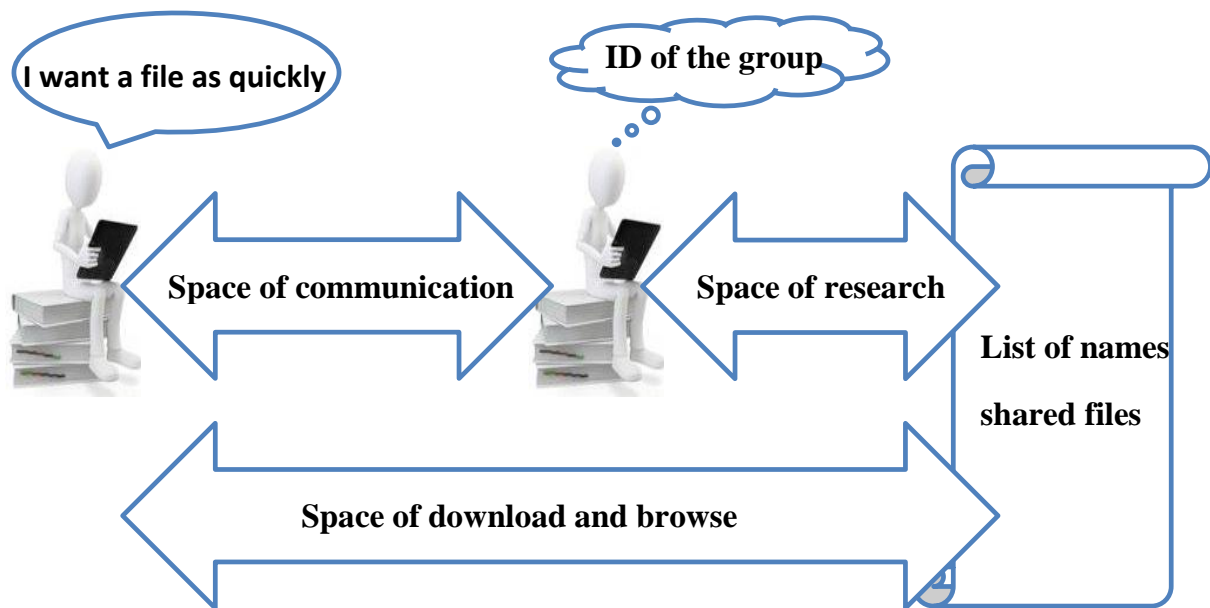


Figure 9- Exposition of the problem

The peer "student" is the part "visible" and useful in our system. This is the application allowing share and downloads files.

Before downloading, a peer must satisfy certain constraints:

- It is necessary to be a member in a peer group via a predefined ID.
- He must share files.
- This quantitative criterion is not qualitative, he is quite possible to share a single file, with no content. He must connect to the group, and inform him of its list of shared files.

### 4. Architecture of the system:

In order to satisfy needs of users of our system, the services of this last must perform the following functions.

- 1- **Sharing:** This service allows to share files between one or more groups of students who have the same identifier.

- 2- **Search, Download and Browse:** these services allows you to search any file to the network announced, then the user can see and start to download selected files.
- 3- **Chat:** View the peers connected to networks and allowto send and receive messages.

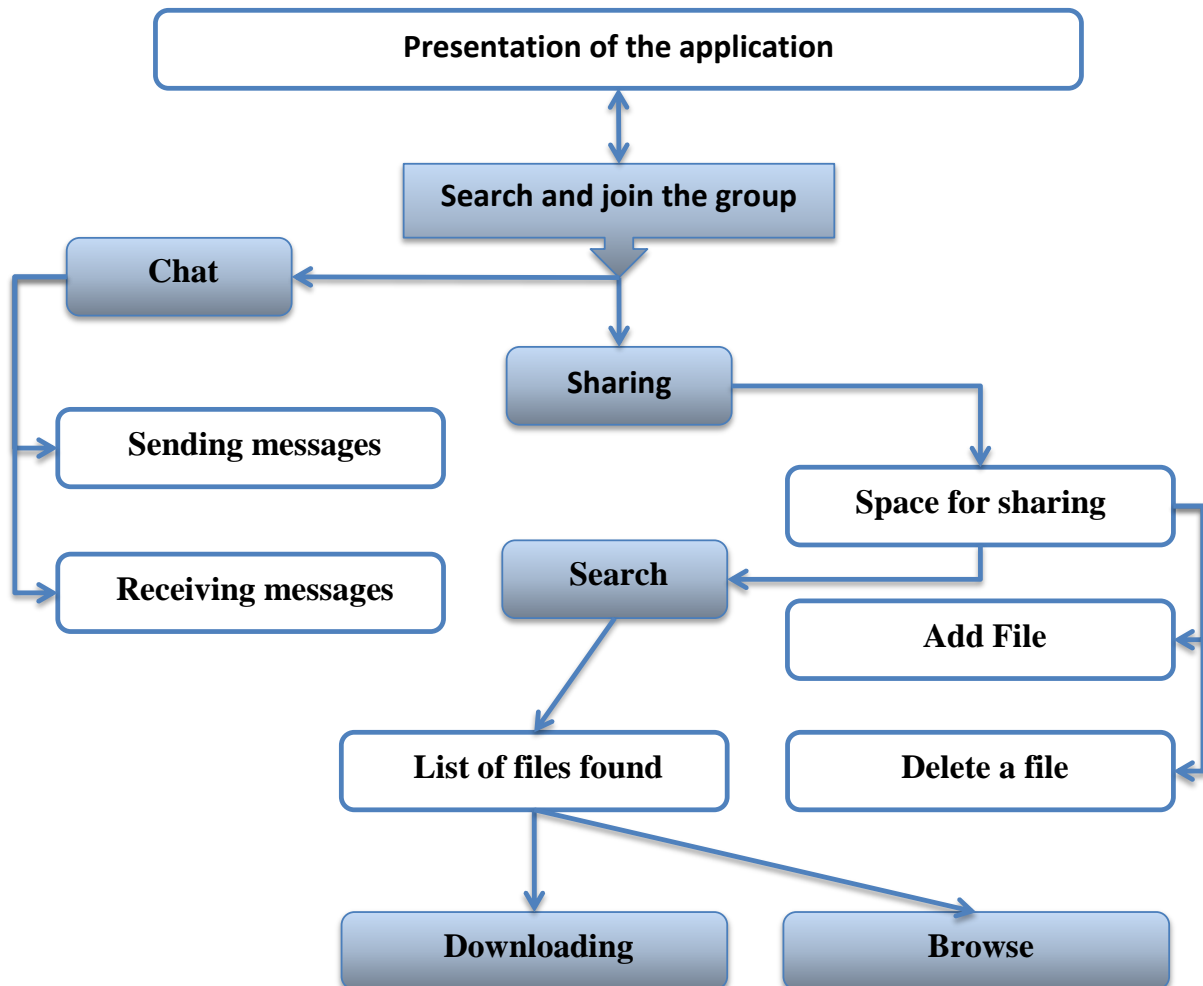


Figure 10- General architecture of the system

## 5. Design and modeling :

We will proceed in this part the modeling and the design of our application. One of the best ways to model how an application works is using the UML language (Universal Modeling Language).

To modelize our application, we use the UML use casediagram, class diagram and sequence diagram. The choice of these diagrams is justified as long

as they are sufficient to cover all the functional and relational aspects of our application.

## 5.1. Functional modeling:

### 5.1.1. The use case of the system:

Use cases can structure user requirements and the related objectives of a system. They focus the expression of system requirements on its users.

According to use cases, we can deduce several scenarios whose determination depends mainly on what the user expects the system. Use cases identify system users (actors) and their interaction with the system.

### 5.1.2. Description of the diagram :

The figure below illustrates the complete scenario of full utilization of the system. We included events that are transparent to the peer to better model application.

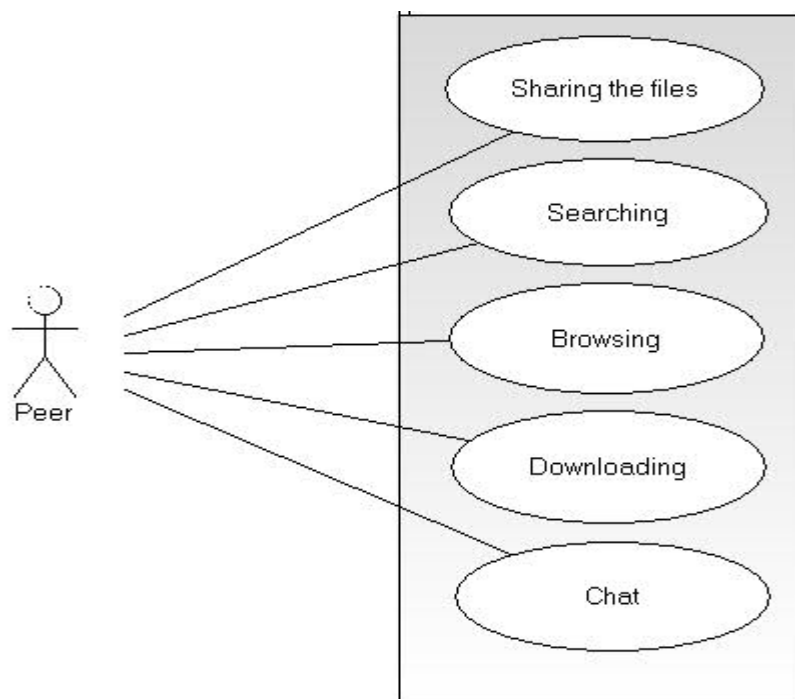


Figure 11- Use case diagram

The scenario presented by the figure is as follows:

- The peer starts the application and network enabled.

- The peer sends a file shared on the P2P network that is making available another peer for the search, downloads and browse.
- The peer has a list of shared files. Can search using the name of the file.
- Once the file found, the peer sees this file and launches the procedure of downloading.
- This system offers the communication between the peer across a chat service.

## 5.2. The static modeling :

### 5.2.1 Class diagram :

The class diagrams express in general the static structure of a system, in terms of classes and relations between these classes. As well as a class describes a set of objects, an association describes a set of links, the objects are instances of classes and relationships are instances of instances of relations.

In the following paragraphs we will present the classes used for the users of our application namely "Peer", "PlayListManager", "Media", "Multicadting" and the corresponding class diagrams.

### 5.2.2 Description of the diagram:

**The class Peer:** on this class unfolds the procedures "sharing, search, download" between different peer.

**The class Multicadting:** It makes available connection services and group.

**The class PlayListManager:** it initializes the list of announcements and the list of shared files.

**The class Media:** they contain all the properties of each shared file.

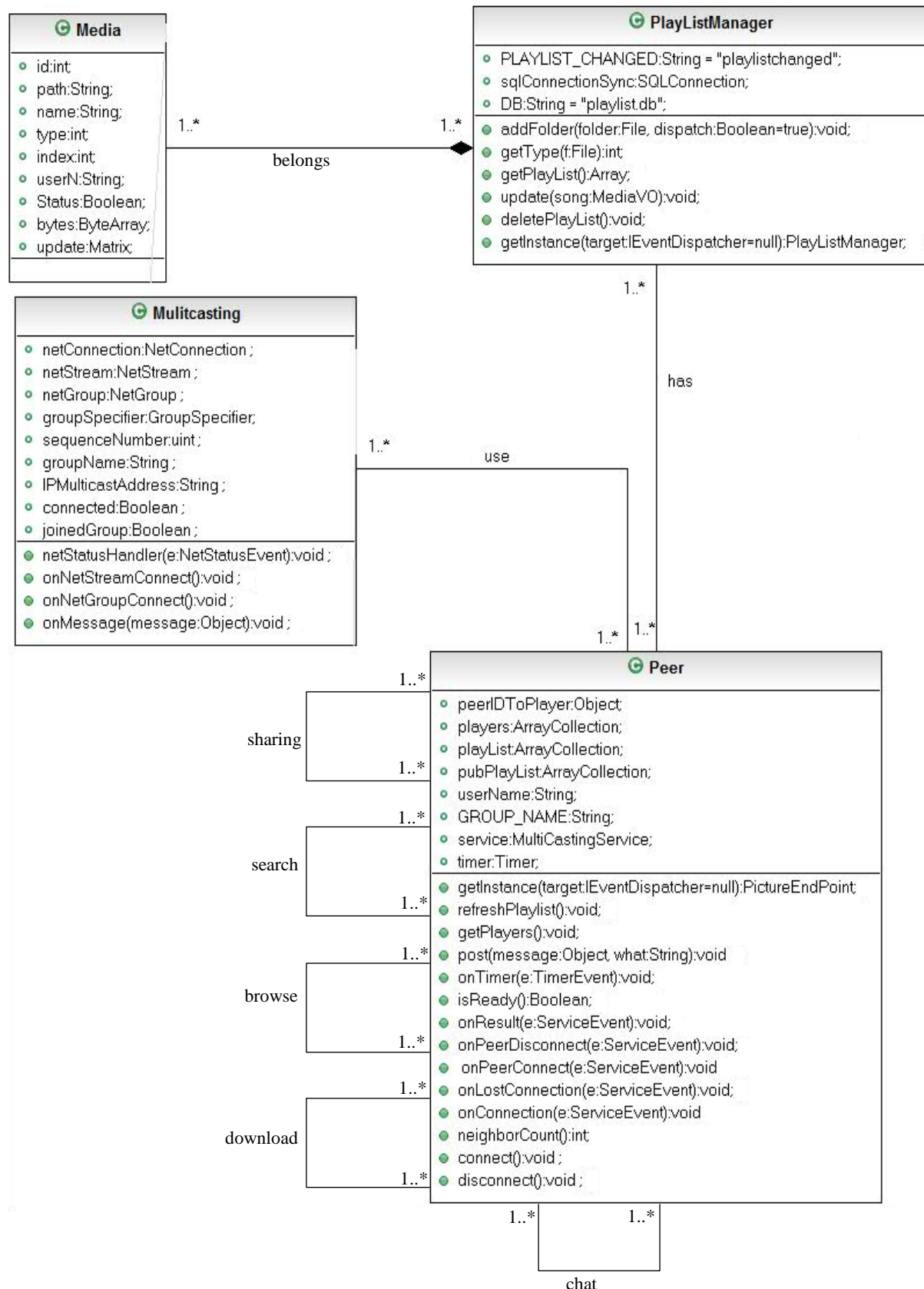


Figure 12- Class diagram

### 5.3. Dynamic modeling:

It models the operation of the system by describing all its interactions.

#### 5.3.1. The sequence diagram

The sequence diagram illustrates the operation of the system over time. It allows to analyze the chronological progress of the interactions between objects in the system to better understand its internal functioning.

#### The sequence diagram of sharing:

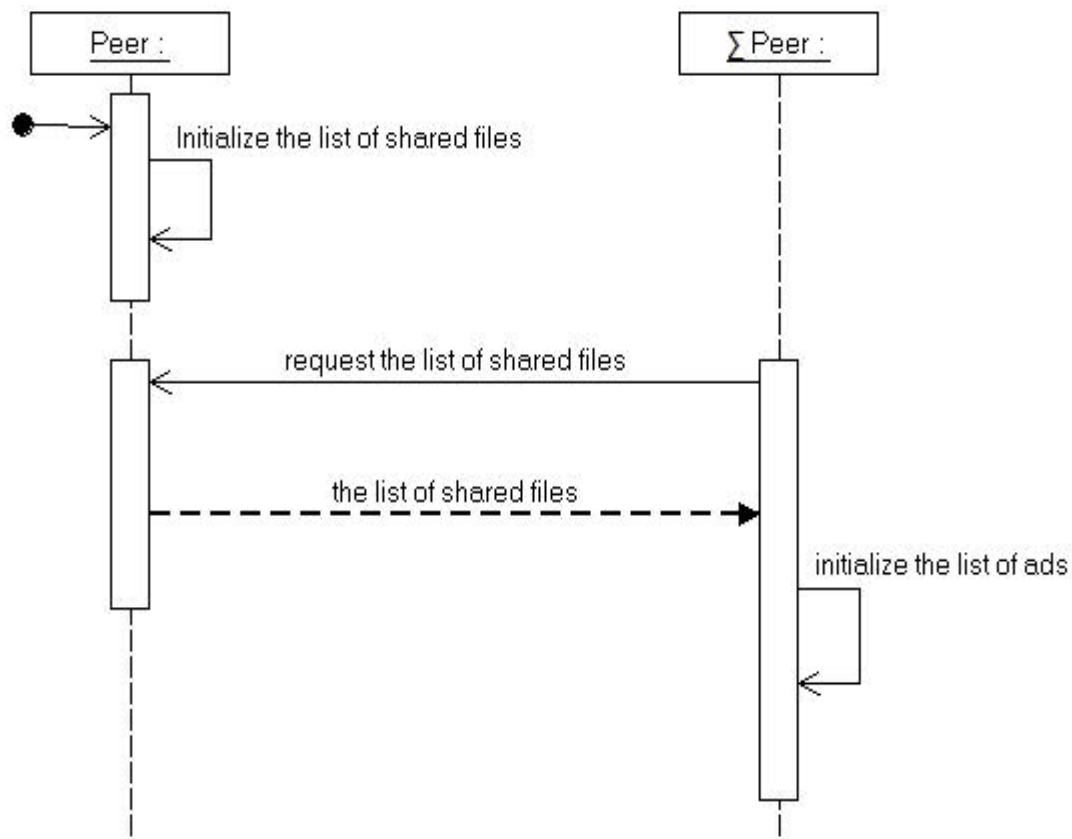


Figure 13- The sequence diagram of sharing

When connecting of peer to the network, sharing follows the following order:

- The peer Initialized the list of shared files.
- The other peer requesting lists of names of shared files.
- The peer sends the list of names of shared files.
- The other peer initializethe list of announcements.



### The sequence diagram of the search, download and browse:

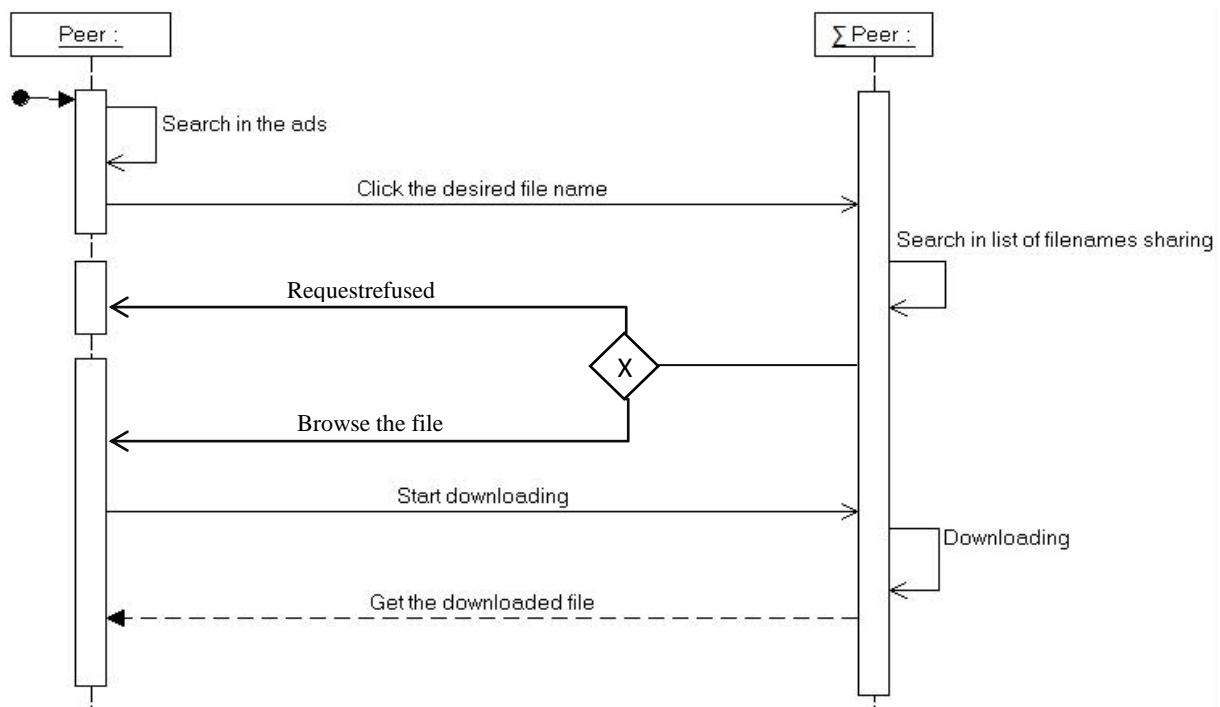


Figure 14- The sequence diagram of the search, download and browse

This sequence diagram explains the task of research from the list of shared files including the download task and browse task.

The pair looks for the file sharing network. Once the file is found, click on the file in order to browse and download.

### Thesequence diagram of the Chat:

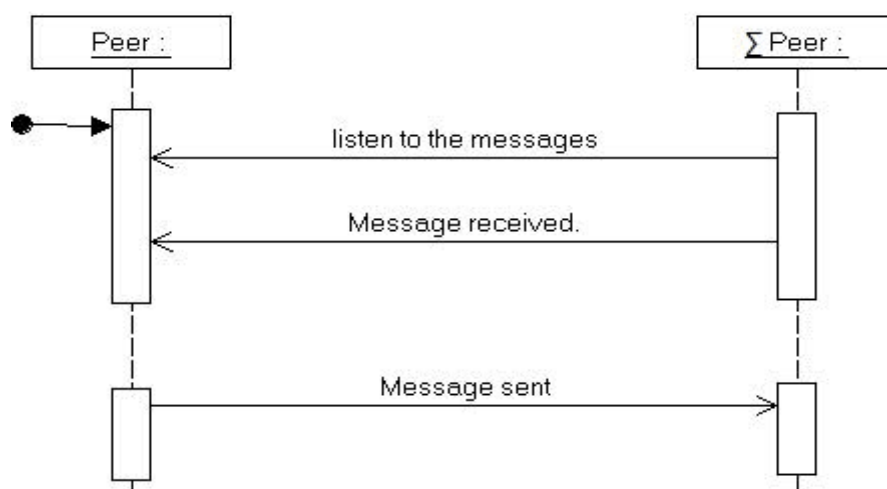


Figure 15- Sequence diagram of the Chat

The mechanism of chat works as follows: The peer is ready to listen to messages emits for other peers, It also sends messages.

### 5.3.2. State-transition diagram:

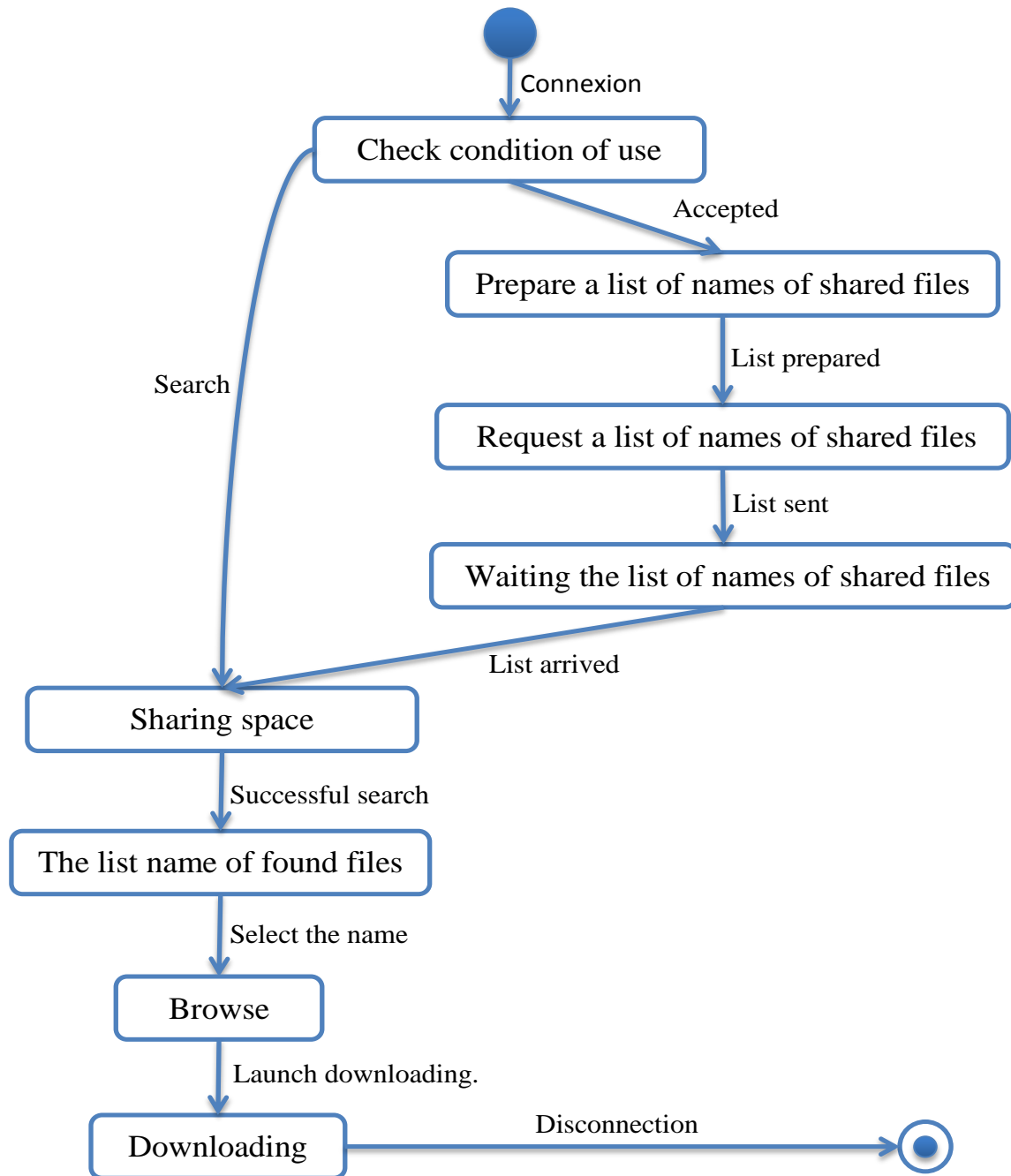


Figure 16- State-transition diagram

## 6. Conclusion:

The design study that unfolds in this chapter, placed as the most important task in any life cycle of software, this is why it takes a large percentage of time developing.

In our work we have modeled this study by a simple and effective tool which is the UML to better understand all the procedures that exist in this project and their relationships.

As the conceptual study has directly influenced the successful implementation of system, in the next chapter we went to the interpretation of this modeling on a development environment.