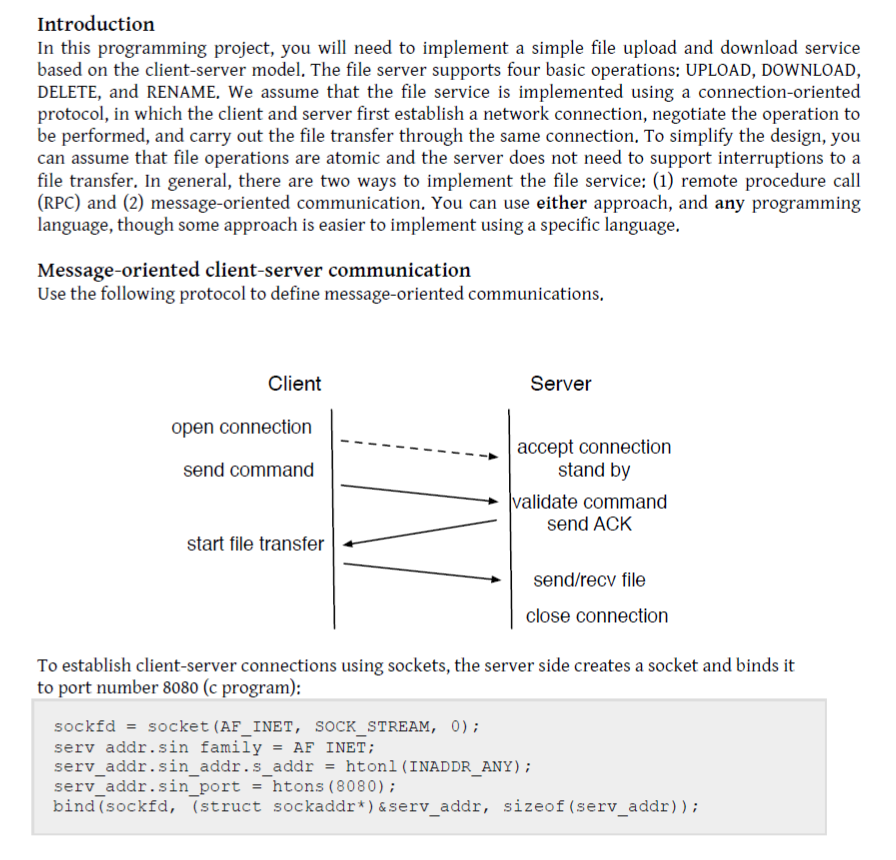
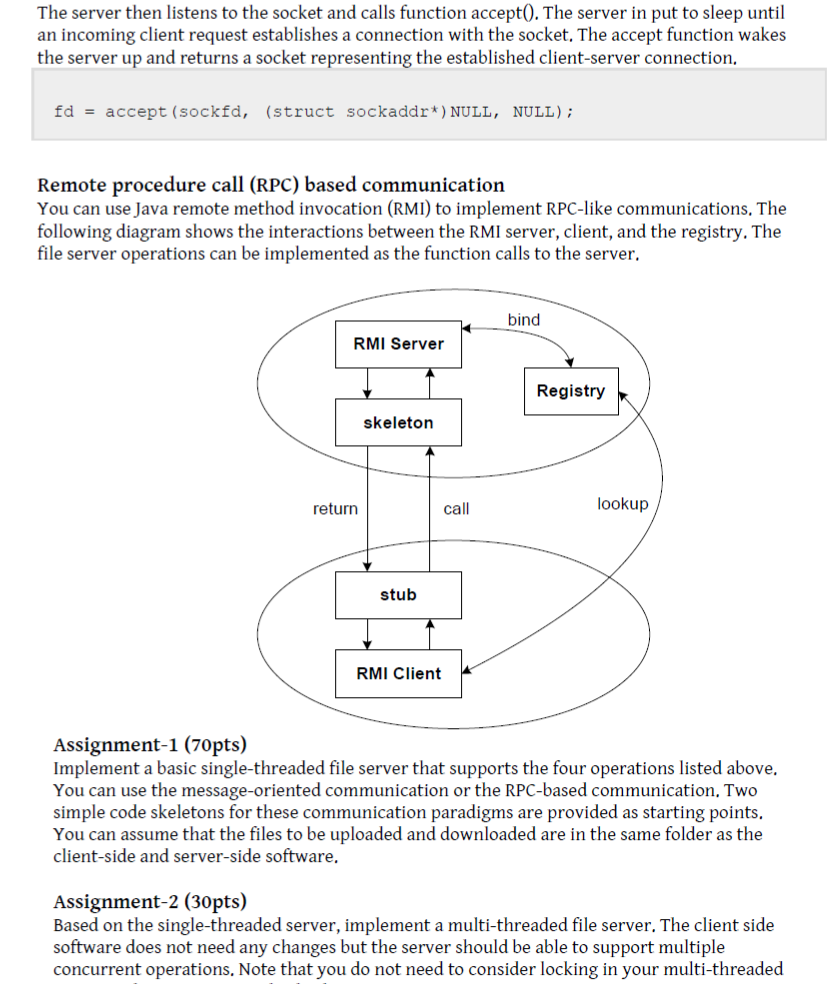
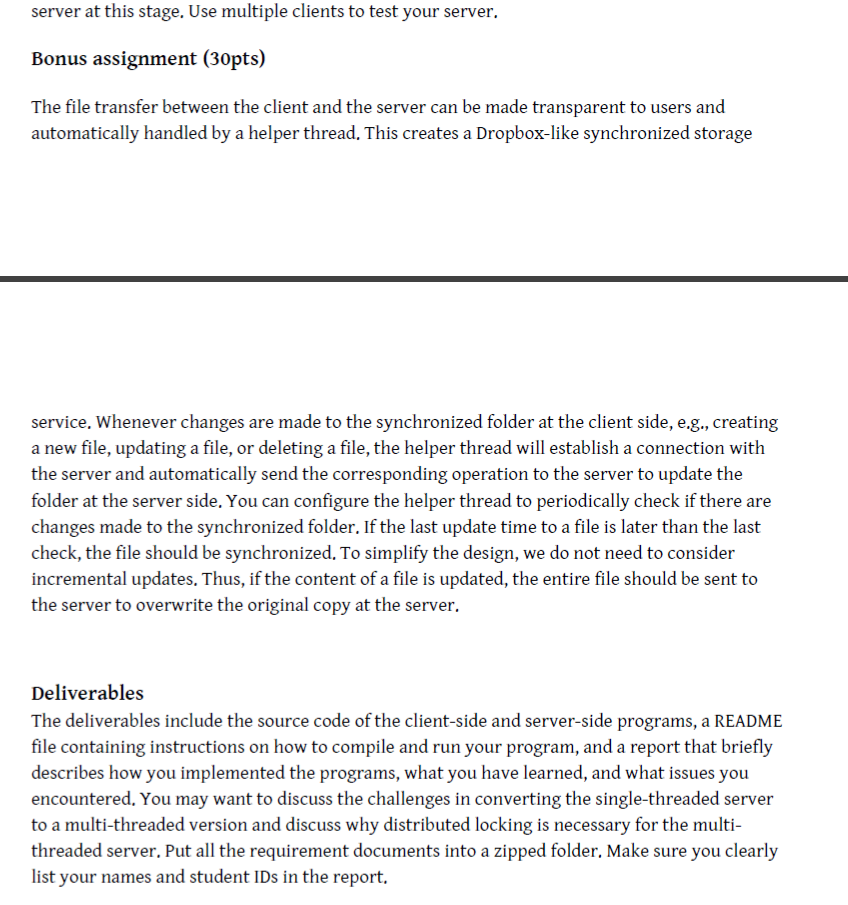
**Problem:**







**Solution:**

**Project Content:**

Assignment1 Folder contains files:

1. MultithreadedServerSocket.java
2. Client.java
3. ClientServerThread.java

Assignment2 Folder contains files:

1. Server.java
2. Client.java

**Instructions to compile the programs:**

Unzip the content into a directory.

Assignment 1:

Type the following command in the terminal

***javac MultithreadedServerSocket.java Client.java ClientServerThread.java***

Assignment 2:

Type the following command in the terminal

***javac Server.java Client.java***

**Instructions to run the program:**

Open at least two terminals: one as a server and other as a client.

Assignment 1:

At Terminal 1, type the following command in the terminal

***Java MultithreadedServerSocket.java***

At Terminal 2, type the following command in the terminal

***Java Client.java***

Terminal 2 will now ask you for input.

Press 1 to Upload File, 2 to Download File, 3 to Rename File and 4 to Delete File.

For each input, the Server will prompt the user to provide the filename to handle the action with. If the filename is input.txt, type:

***input***

For rename action, the new filename should also be provided. If the file to be renamed is input.txt and it is to renames as newInput.txt, then type:

***input newInput***

*NOTE: Do not add the .txt extension to any of the filenames through this project execution*

The inputs are sent to the Server for further processing and the appropriate actions are done.

Assignment 2:

At Terminal 1, type the following command in the terminal

***Java Server.java***

At Terminal 2, type the following command in the terminal

***Java Client.java***

Terminal 2 will now ask you for input.

If you want to upload a file named filename.txt, type the following:

***1 filename***

*NOTE: Do not add the .txt extension to any of the filenames through this project execution.*

If you want to download a file named filename.txt, type the following:

2 filename

If you want to rename a file named filename.txt to renameTo.txt, type the following:

3 filename renameTo

If you want to delete a file named filename.txt, type the following:

4 filename

The inputs are sent to the Server for further processing and the appropriate actions are done.

This program can handle multiple clients simultaneously.

**Description:**

We have implemented the program using Sockets. The single-threaded Server and Client program simply exchanges two inputs and outputs respectively. The multi-threaded Server and Client program was implemented using Threads to support one to many relationship for Server and client respectively.

**What we learnt:**

We learnt how to create independent Client and Server using Socket programming.

We also learnt how to perform multithreading using Java. Also that locking is important while working with distributed systems.

**Issues encountered:**

We had to implement threads in Java to support one to many relationships, which was complicated.

One significant issue was the global variables, which are global to thread but not global to the entire program.

**Significance of locking:**

Distributed locking is required for the following reasons:

When one client is using a file to perform an action, say rename, and another client wants to access the dame file for the same action, it would create issues.

Hence distributed locking is required when one person is working with a file, such that the other clients who wish to use that file will wait until the file is unlocked.