Final Project Report

Name: Xinyu Wang, Yongle Lin, Muhan Na

Project Description

This project is a simple version of a distributed storage system. Users can create, update and share a file in a remote storage.

The main functions of this project include:
- A user can register and login his/her account
- A user can check what file he/she owns
- A user can create a file
- A user can update a file's content
- A user can share a file with another user
- A user can get notification about a file that he/she owns

Project Component

Client:

Client.java provides the main interface for the client to communicate with the service. Client end's responsibilities include:
- Display instruction for users
- Convert users' commands into unified message used to communicate with server
- Display response got from server about request process
- Management receivers of notification about files the user owns

Server:

ServerNode.java provides the main interface to process ClientMessage sent from client end. According to the different types of requests represented by the ClientMessage, different services will be invoked. There are four services used to take care of client requests.

Account service:

Account service provides service related to a user account. Responsibilities of account service include:
- User account registration
- User account log in
- User's file list display

File service:

File service provides service related to a file. Responsibilities of file service include:
- Create a file with title and content
- Update a file content
- Open a file to check content

Share service:
- Share a file with a user by his/her username

In the implementation, the user who is shared this file will add the file ID in his/her file list. This user will also be added into the list of file's owner.

All the services to update databases are using PAXOS.

Notification service:
In this project, HornetQ Server is used to provide the JMS messaging service. The server is configured in class NotificationHQServer.java. Before running the project, this class should be started along with Client.java and ServerNode.java.

Our notification service uses point-to-point messaging domain, which means queue is used instead of topic. We have several message queues registered in hornetq-jms.xml and they are named by usernames. When there is a notification about a file related to a user, notification service will send a notification message to the queue named by this file owner's username.

At the client end, when there is connection established between a user and a file, the client end will start a receiver for this file. Such connection can be established when a user creates a file or he/she accepts a share invitation for a file.
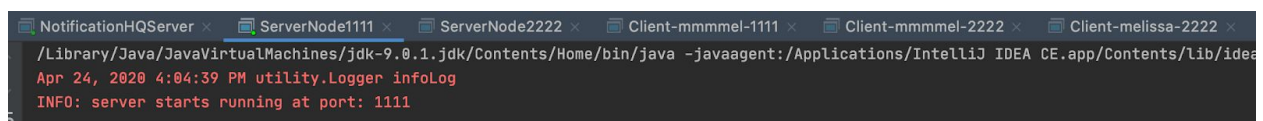
Running Result
A demo video is available for this flow.

Start with
NotificationHQServer.java
ServerNode.java <port number> <peer port>
Client.java localhost <port number>

Register a user "mmmmel" on port 1111:

The user "mmmmel" creates a new file:

The user "mmmmel" can also log in on port 2222 and get the file:

```
   NotificationHQServer ×    ServerNode1111 ×    ServerNode2222 ×    Client-mmmmel-1111 ×    Client-mmmmel-2222 ×    Client-melissa-2222 ×
   OPEN <file ID>: open your file to view the content
   CREATE <file title> <file content>: create a file with a title and content
   SHARE <file ID> <username>: share a file with a user
   UPDATE <file ID> <new content>: update content for a file

   Apr 24, 2020 4:06:49 PM utility.Logger infoLog
   INFO: Listening to mmmmel
   LIST
   Apr 24, 2020 4:07:16 PM utility.Logger infoLog
   INFO: Sending request: LIST
   SUCCESS to LIST
   File ID: 100004 Title: title1   Version: 1
```

Then user "mmmmel" wants to share a file with another user "melissa":


```
   NotificationHQServer ×    ServerNode1111 ×    ServerNode2222 ×    Client-mmmmel-1111 ×    Client-mmmmel-2222 ×    Client-melissa-2222 ×
   Apr 24, 2020 4:05:48 PM utility.Logger infoLog
   INFO: Sending request: CREATE title1 content1
   SUCCESS to CREATE
   LIST
   Apr 24, 2020 4:05:55 PM utility.Logger infoLog
   INFO: Sending request: LIST
   SUCCESS to LIST
   File ID: 100004 Title: title1   Version: 1
   SHARE 100004 melissa
   Apr 24, 2020 4:09:10 PM utility.Logger infoLog
   INFO: Sending request: SHARE 100004 melissa
   SUCCESS to SHARE
```

Later, the user "melissa" logged in:


```
   NotificationHQServer ×    ServerNode1111 ×    ServerNode2222 ×    Client-mmmmel-1111 ⊗    Client-mmmmel-2222 ×    Client-melissa-2222 ×
   /Library/Java/JavaVirtualMachines/jdk-9.0.1.jdk/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.j
   Welcome!
   Please enter "login" or "signup":
   login
   Please enter your username to log in:
   melissa
   Apr 24, 2020 4:09:24 PM utility.Logger infoLog
   INFO: Successfully logged in.
   Retrieving your files...
   File ID: 100001 Title: title1   Version: 1
   File ID: 100002 Title: title2   Version: 1
   File ID: 100004 Title: title1   Version: 1
   Please type your request...
   LIST: list your files
   OPEN <file ID>: open your file to view the content
   CREATE <file title> <file content>: create a file with a title and content
   SHARE <file ID> <username>: share a file with a user
   UPDATE <file ID> <new content>: update content for a file

   Apr 24, 2020 4:09:24 PM utility.Logger infoLog
   INFO: Listening to melissa
   Message is arrived: File 100004 is shared with you
```

She got a message for the sharing for File 100004. She can list her files and File 100004 is under her account:

```
  NotificationHQServer ×    ServerNode1111 ×    ServerNode2222 ×    Client-mmmmel-1111 ×    Client-mmmmel-2222 ×    Client-melissa-2222 ×
    UPDATE <file ID> <new content>: update content for a file

    Apr 24, 2020 4:09:24 PM utility.Logger infoLog
    INFO: Listening to melissa
    Message is arrived: File 100004 is shared with you
    LIST
    Apr 24, 2020 4:11:13 PM utility.Logger infoLog
    INFO: Sending request: LIST
    SUCCESS to LIST
    File ID: 100001 Title: title1    Version: 1
    File ID: 100002 Title: title2    Version: 1
    File ID: 100004 Title: title1    Version: 1
```

Then the user "mmmmel" can view the content of the file:


```
  NotificationHQServer ×    ServerNode1111 ×    ServerNode2222 ×    Client-mmmmel-1111 ×    Client-mmmmel-2222 ×    Client-melissa-2222 ×
    Apr 24, 2020 4:09:10 PM utility.Logger infoLog
    INFO: Sending request: SHARE 100004 melissa
    SUCCESS to SHARE
    Apr 24, 2020 4:09:10 PM utility.Logger infoLog
    INFO:
    OPEN 100004
    Apr 24, 2020 4:12:02 PM utility.Logger infoLog
    INFO: Sending request: OPEN 100004
    SUCCESS to OPEN
    content1
```

User "mmmmel" updates the file's content:


```
  NotificationHQServer ×    ServerNode1111 ×    ServerNode2222 ×    Client-mmmmel-1111 ×    Client-mmmmel-2222 ×    Client-melissa-2222 ×
    INFO: Sending request: OPEN 100004
    SUCCESS to OPEN
    content1
    UPDATE 100004 content2
    Apr 24, 2020 4:12:40 PM utility.Logger infoLog
    INFO: Sending request: UPDATE 100004 content2
    SUCCESS to UPDATE
    Apr 24, 2020 4:12:41 PM utility.Logger infoLog
    INFO:
    OPEN 100004
    Apr 24, 2020 4:12:49 PM utility.Logger infoLog
    INFO: Sending request: OPEN 100004
    SUCCESS to OPEN
    content2
```

User "mmmmel" on port 2222 can receive a notification of the file update and she
checks the updated content:


```
  NotificationHQServer ×    ServerNode1111 ×    ServerNode2222 ×    Client-mmmmel-1111 ×    Client-mmmmel-2222 ×    Client-melissa-2222 ×
    UPDATE <file ID> <new content>: update content for a file

    Apr 24, 2020 4:06:49 PM utility.Logger infoLog
    INFO: Listening to mmmmel
    LIST
    Apr 24, 2020 4:07:16 PM utility.Logger infoLog
    INFO: Sending request: LIST
    SUCCESS to LIST
    File ID: 100004 Title: title1    Version: 1
    Message is arrived: File 100004 is updated
```

```
INFO: Sending request: LIST
SUCCESS to LIST
File ID: 100004 Title: title1   Version: 1
Message is arrived: File 100004 is updated
OPEN 100004

Apr 24, 2020 4:15:59 PM utility.Logger infoLog
INFO: Sending request: OPEN 100004
SUCCESS to OPEN
content2
```

User "melissa" who is sharing the File 100004 also received a notification and she checked the updated content:


```
UPDATE <file ID> <new content>: update content for a file

Apr 24, 2020 4:09:24 PM utility.Logger infoLog
INFO: Listening to melissa
Message is arrived: File 100004 is shared with you
LIST
Apr 24, 2020 4:11:13 PM utility.Logger infoLog
INFO: Sending request: LIST
SUCCESS to LIST
File ID: 100001 Title: title1   Version: 1
File ID: 100002 Title: title2   Version: 1
File ID: 100004 Title: title1   Version: 1
Message is arrived: File 100004 is updated
```



```
File ID: 100001 Title: title1   Version: 1
File ID: 100002 Title: title2   Version: 1
File ID: 100004 Title: title1   Version: 1
Message is arrived: File 100004 is updated
OPEN 100004
Apr 24, 2020 4:17:02 PM utility.Logger infoLog
INFO: Sending request: OPEN 100004
SUCCESS to OPEN
content2
```

If now user "melissa" logged out and user "mmmmel" updated File 100004 again:


```
Apr 24, 2020 4:12:49 PM utility.Logger infoLog
INFO: Sending request: OPEN 100004
SUCCESS to OPEN
content2
UPDATE 100004 NEWCONTENT
Apr 24, 2020 4:17:45 PM utility.Logger infoLog
INFO: Sending request: UPDATE 100004 NEWCONTENT
SUCCESS to UPDATE
```

After user "melissa" logged in later, she will receive a notification about the update and she checked it:

```
NotificationHQServer ×   ServerNode1111 ×   ServerNode2222 ×   Client-mmmmel-1111 ×   Client-mmmmel-2222 ×   Client-melissa-2222 ×
/Library/Java/JavaVirtualMachines/jdk-9.0.1.jdk/Contents/Home/bin/java -javaagent:/Applications/IntelliJ IDEA CE.app/Contents/lib/idea_rt.ja
Welcome!
Please enter "login" or "signup":
login
Please enter your username to log in:
melissa
Apr 24, 2020 4:17:58 PM utility.Logger infoLog
INFO: Successfully logged in.
Retrieving your files...
File ID: 100001 Title: title1   Version: 1
File ID: 100002 Title: title2   Version: 1
File ID: 100004 Title: title1   Version: 1
Please type your request...
LIST: list your files
OPEN <file ID>: open your file to view the content
CREATE <file title> <file content>: create a file with a title and content
SHARE <file ID> <username>: share a file with a user
UPDATE <file ID> <new content>: update content for a file

Apr 24, 2020 4:17:58 PM utility.Logger infoLog
INFO: Listening to melissa
Message is arrived: File 100004 is updated
```

```
NotificationHQServer ×   ServerNode1111 ×   ServerNode2222 ×   Client-mmmmel-1111 ×   Client-mmmmel-2222 ×   Client-melissa-2222 ×
LIST: list your files
OPEN <file ID>: open your file to view the content
CREATE <file title> <file content>: create a file with a title and content
SHARE <file ID> <username>: share a file with a user
UPDATE <file ID> <new content>: update content for a file

Apr 24, 2020 4:17:58 PM utility.Logger infoLog
INFO: Listening to melissa
Message is arrived: File 100004 is updated
OPEN 100004
Apr 24, 2020 4:20:53 PM utility.Logger infoLog
INFO: Sending request: OPEN 100004
SUCCESS to OPEN
NEWCONTENT
```

Further improvement

User interface:
Currently, the client can only access and send requests using the terminal, in the future, after adding the user interface to our project, it's more convenient and intuitive for users to use this tool.

Format of file:
In this project, for simplicity, a file is represented by a customized class and the content of the file is a simple string. However, in the future, with a UI, we can extend to larger real files.

Online file editing:
In this project, we only support users to update and save the change to a file in one time, which means that files owners can't share their process while editing the file. In the future we could allow multiple users to edit the same file at the same time.

Notification service:

During the implementation of notification service, a limitation of queues creation is encountered. Since the queue names are registered in hornetq-jms.xml in advance, with files' number growing, we need a dynamic way to coordinate the queues' registration.

Video Demo:

https://www.youtube.com/watch?v=zVNDzdOVfx0&feature=youtu.be