PES University, Bengaluru

Subject: Object Oriented Analysis and Design with Java

Title: Secure File Sharing System

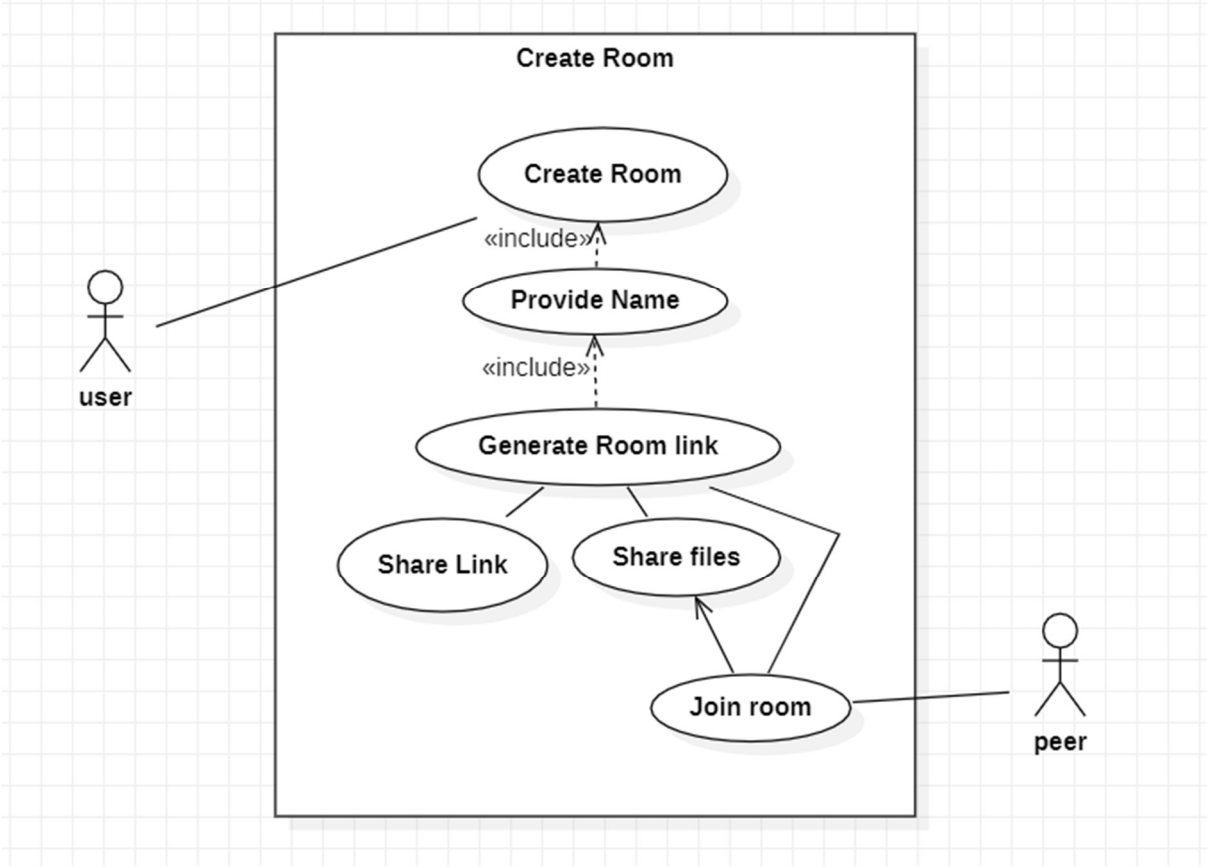
Use Case submission

Team members:

Suraj S Raju	PES2UG20CS812
Chinmay Gowda	PES2UG20CS902
Chandan Kumar S	PES2UG20CS804
Vijay J	PES2UG20CS815

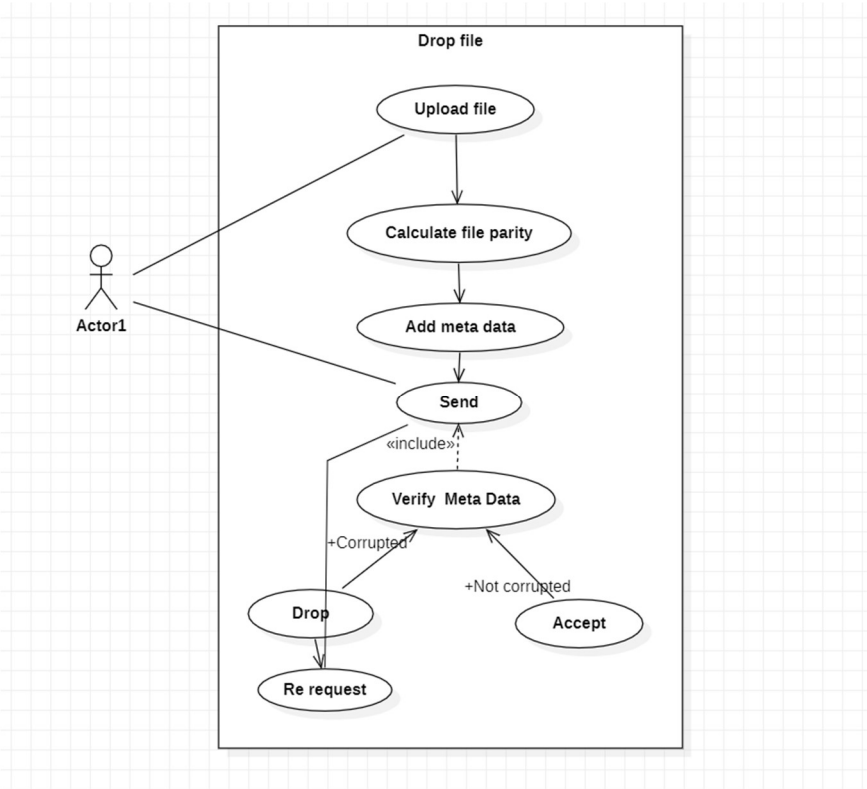
1. By: Chandan Kumar s

(a) CreateRoom Use Case:



1. By: Vijay J

(b) DropFile Use Case:



Use case specifications:

1. Name: CreateRoom
2. Summary:Create a room where peers can share files
3. Actor:Client/Peer
4. Preconditions: The Client has entered the platform
5. **Description:**

- Click on “create room”
- Click on “generate room link”
- Click on “share link” and share it with other peers and the peers can join the room if the room limit has not been exceeded

6. Exceptions: The room limit exceeds
7. Post-condition: Peers can join and share files

1.Name: DropFile

2.Summary: On receiving the file, drop the file if it was corrupted during transfer

3. Actor: Receiver/Peer

3. Preconditions: File is uploaded and is ready to be sent over the network

4. Description:

- Calculate the file parity at the sender side, which upon received at the receiver side is used to verify the integrity of the file

- It is added as metadata of the file

- It is encrypted, sent over the network and received on the other side, and decrypted.

- The metadata appended to the file is used to check the integrity of the file

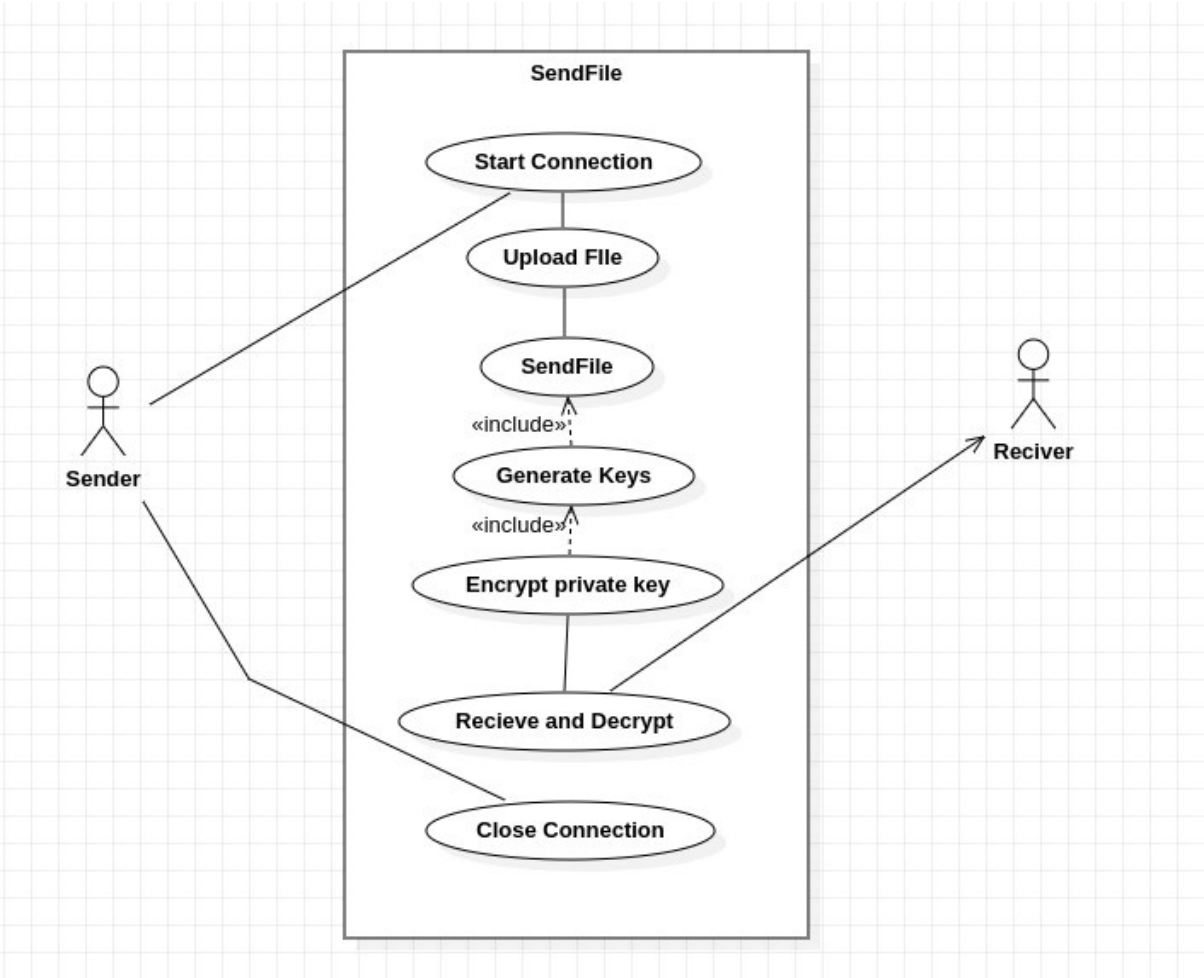
- If the file is found to be tampered, it is dropped immediately and a new request to resend the file goes to the sender, else accept the file.

5. Exceptions: The file doesn't reach the receiver side

6. Post-condition: Receiver can download the files

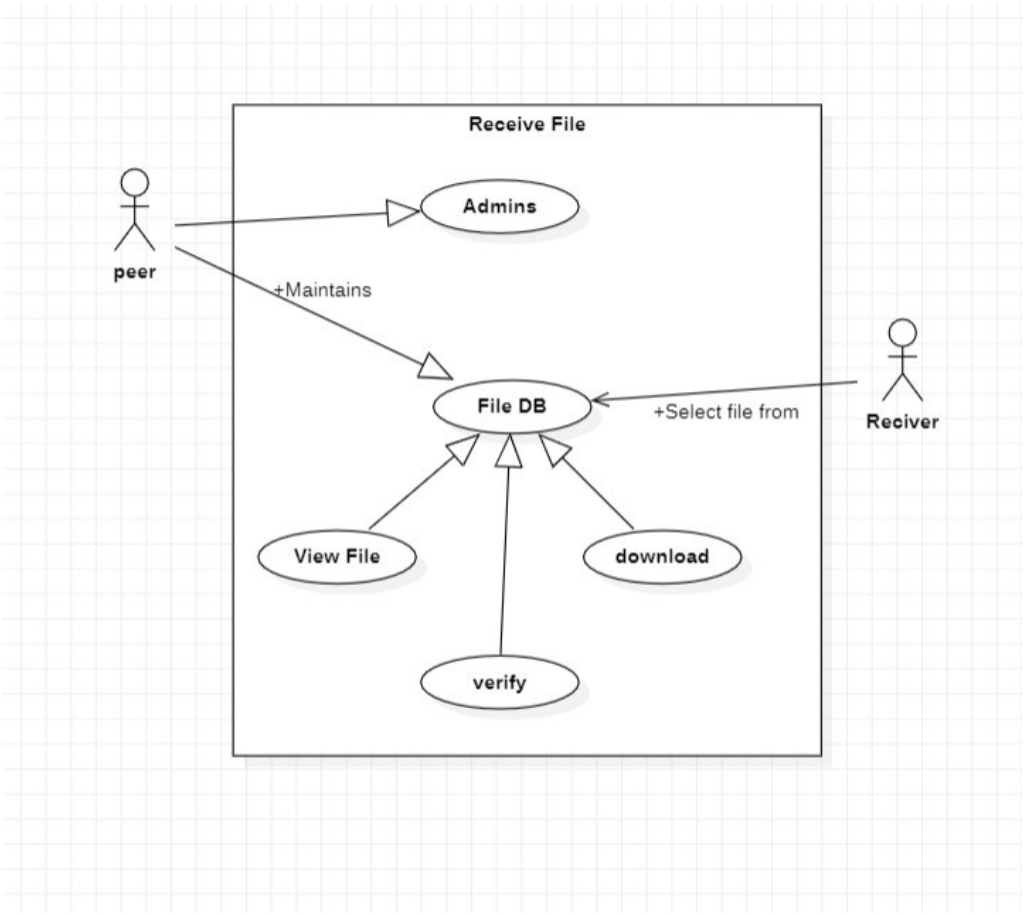
2. By: Chandan Kumar S/ Vijay J

(c) SendFile Use Case:



1. By: Chinmay Gowda

(d) ReceiveFile Use Case:



Use case specifications:

1. Name: SendFile
2. Summary: Prepare the file to be sent over the network
3. Actor: Sender/Peer
4. Preconditions: The file is uploaded onto the platform
5. Description:

- On starting the connection, keys generated for secure transmission i.e encryption
- Share the public key to the receiver
- Encrypt the file with the key already generated
- Send the file to the receiver
- Receive at the receiver side

- Close connection

6. Exceptions: File doesn't exist in the DB

7. Post-condition: Receiver can receive the sent file

2.

1. Name: Receive File

2. Summary: Receive the file sent over the network.

3. Actor:Peer/Receiver

4. Preconditions: The Sender has pushed a file to the fileDB

5. Description:

- Click on "receive" file

- Click on "verify" to verify the contents of the file by decrypting the file using the public key of the sender.

- Click on "download" to download the file after successful verification.

6. Exceptions: Verification fails.

7. Post-condition: The file can now be downloaded by the receiver.

3. By Chinmay Gowda

(e) UploadFile Use Case:

1. Name: Upload File use case

2. Summary: Select the file and upload the file in the FileDB

3. Actor: Client/Peer

4. Pre-conditions: The Client has entered the platform

5. Description:

- Click on "upload" to upload the file

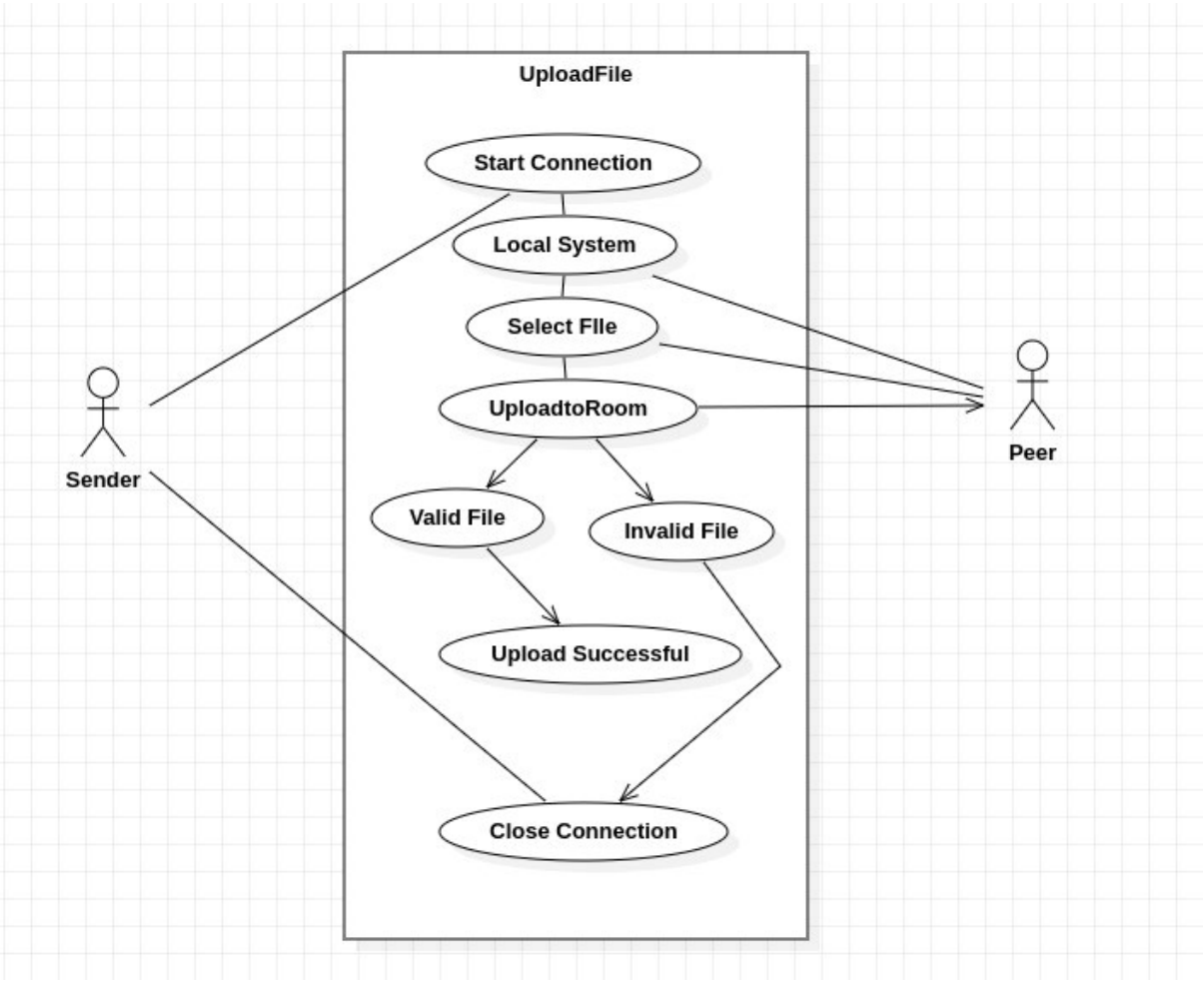
- Select the file from the peer local system.

- Click on "OK" to start uploading.

- The uploading actions have start, pause and stop options.

6. Exception - File is invalid or corrupted.

7. Post-condition - Send the file for encryption.



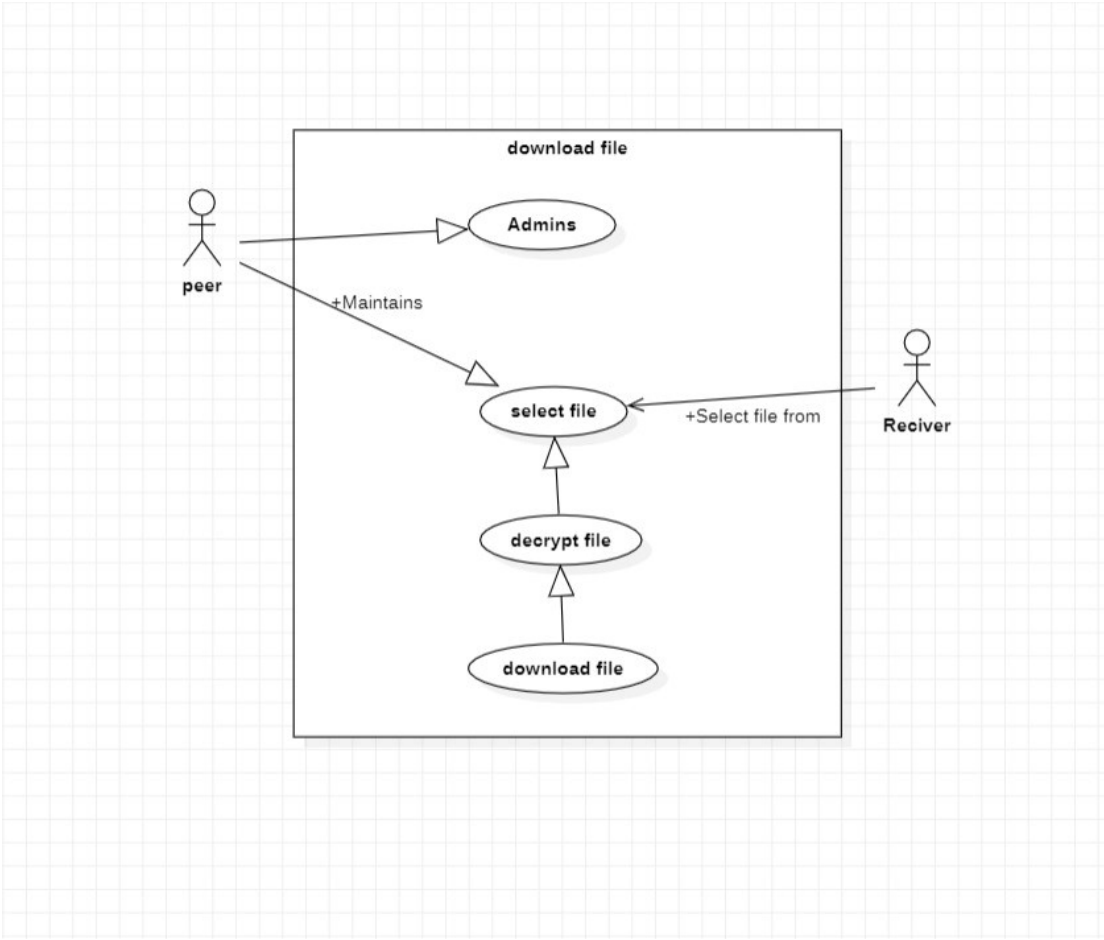
3. By Suraj S Raju

(f) DownloadFile Use Case:

1. Name: Download File use case
2. Summary: Download the shared file in the room
3. Actor: Receiver/Peer
4. Pre-conditions: The file had been received and verified by decryption.
5. Description:

- Click on “download” to download the file.
- Select the destination to download in the peer’s local system.
- Click on “OK” to start downloading.
- The downloading actions have start, pause and stop options.

6. Exception - File is invalid or corrupted (verification failed).
7. Post-condition - The peer can continue being in the room to share and receive files.



File Object

3. By Suraj S Raju