

OOAD ASSIGNMENT

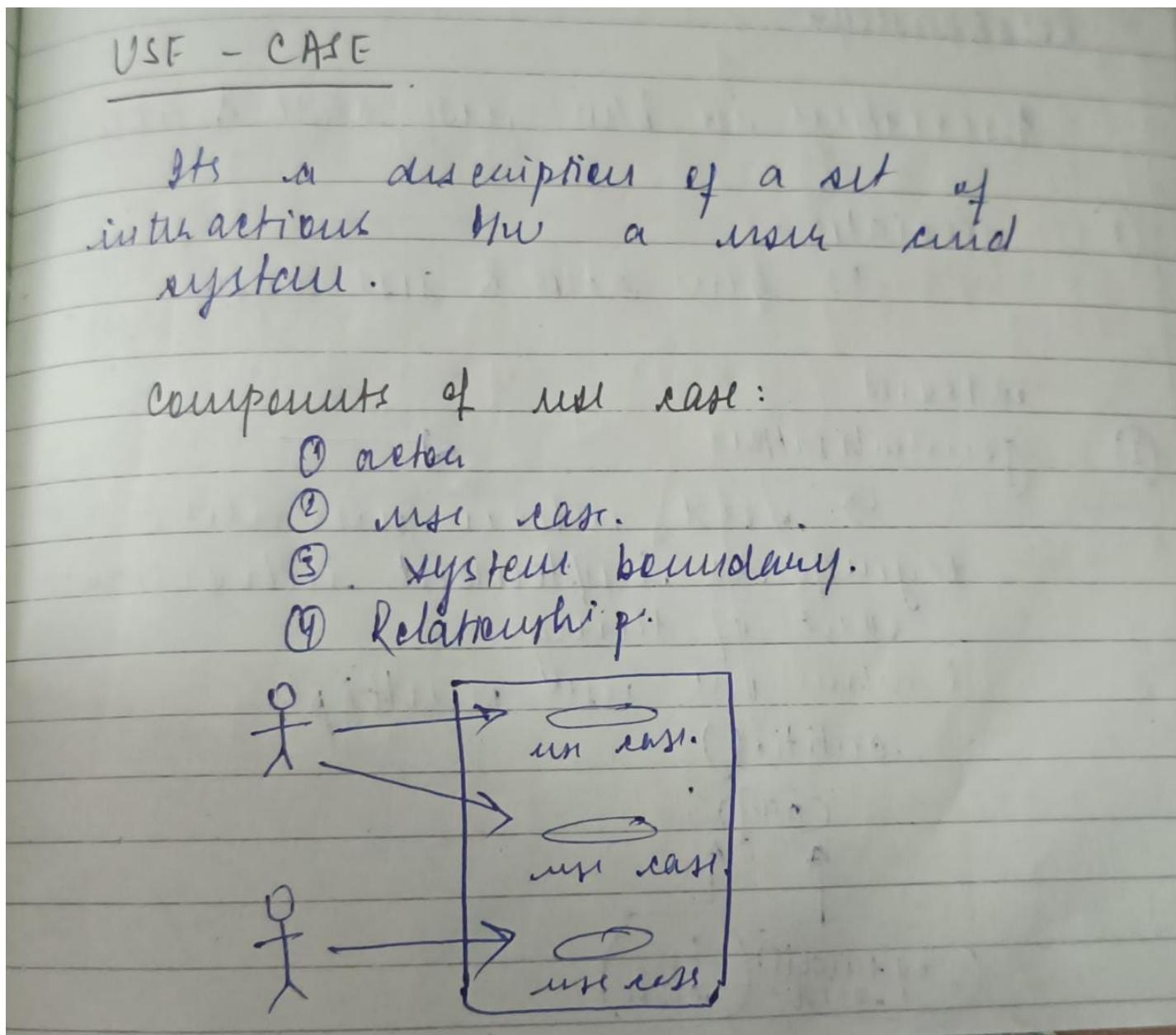
Name - Ishika Kesarwani

Sap ID - 500076373

Roll no. 92

Batch - AIML B-3

1) USE CASE DIAGRAM



- ① Actor:
- human / automated system
 - not part of system.

User can:

- dialogue w/ the actor & system.
- These are sequences of actions that user takes on a system to get particular target.

Relationships

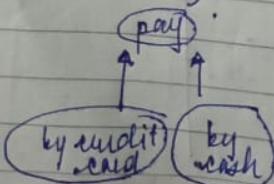
Association b/w user case & actors.

① association

↳ b/w actor & user case.

② generalization
extension

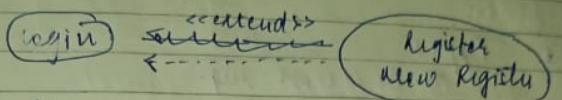
↳ used when we are trying to implement inheritance (when we use multiple entities)



③ use

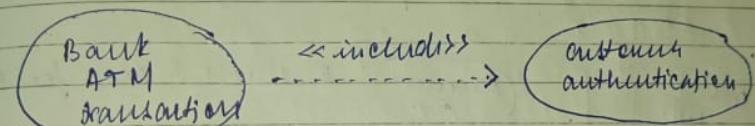
$\leftarrow \rightarrow$

④ extend

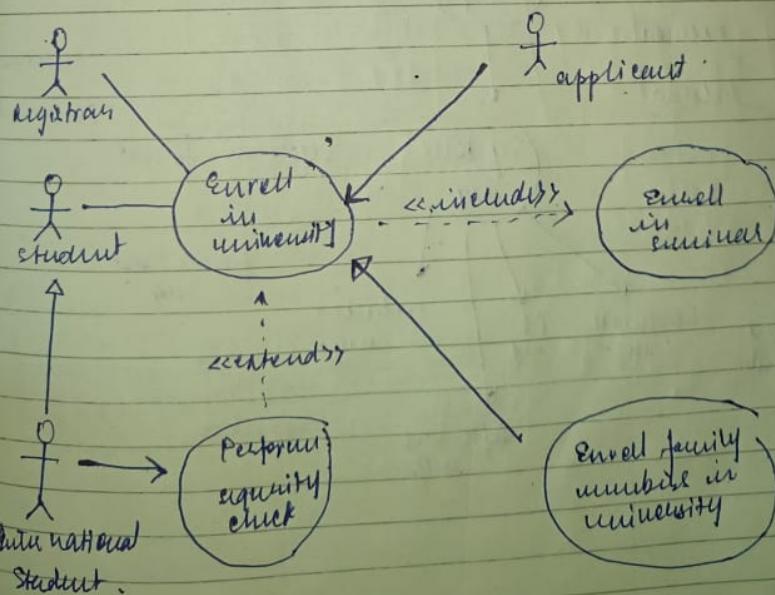


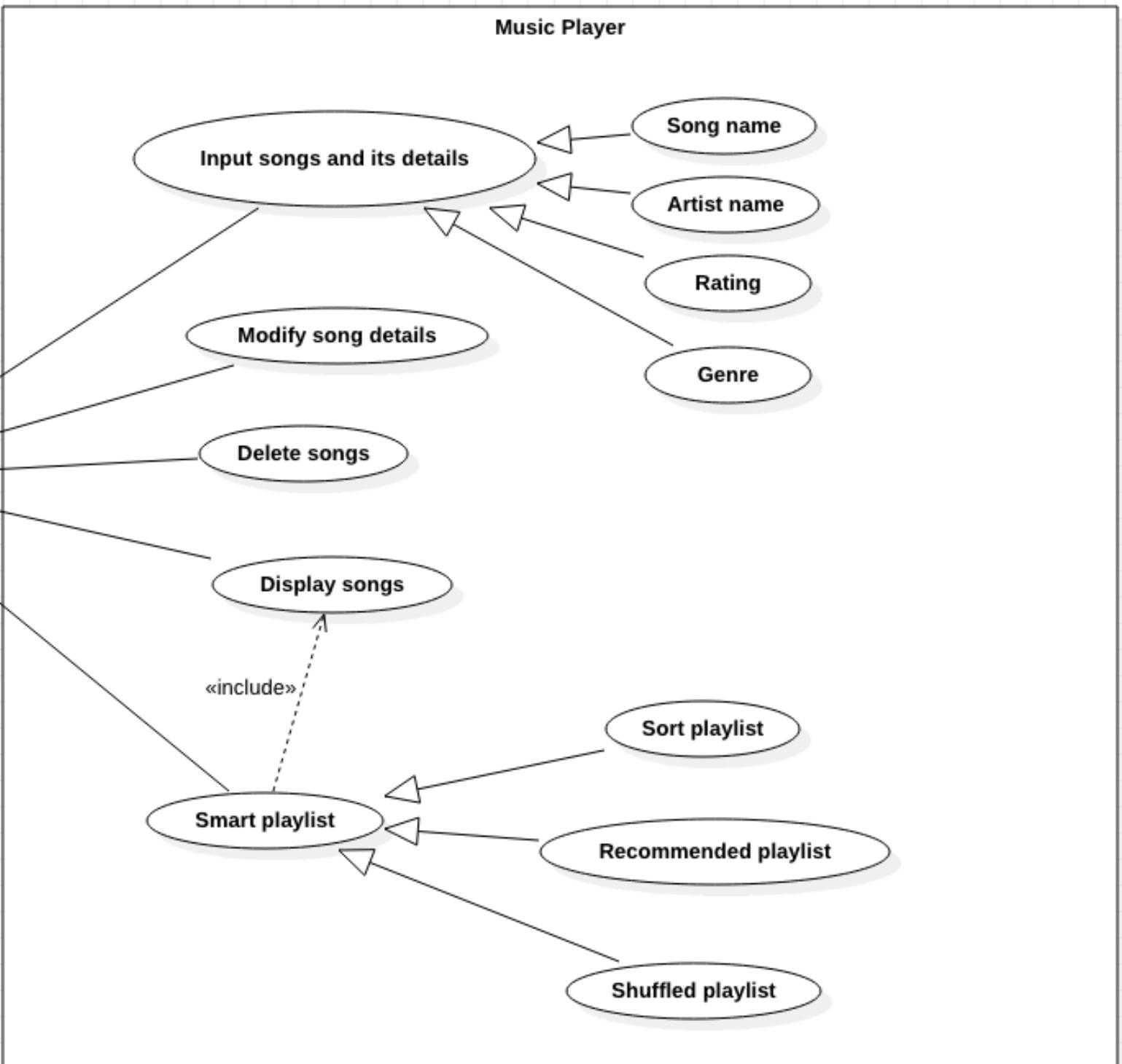
↳ called if something is optional.

⑤ include

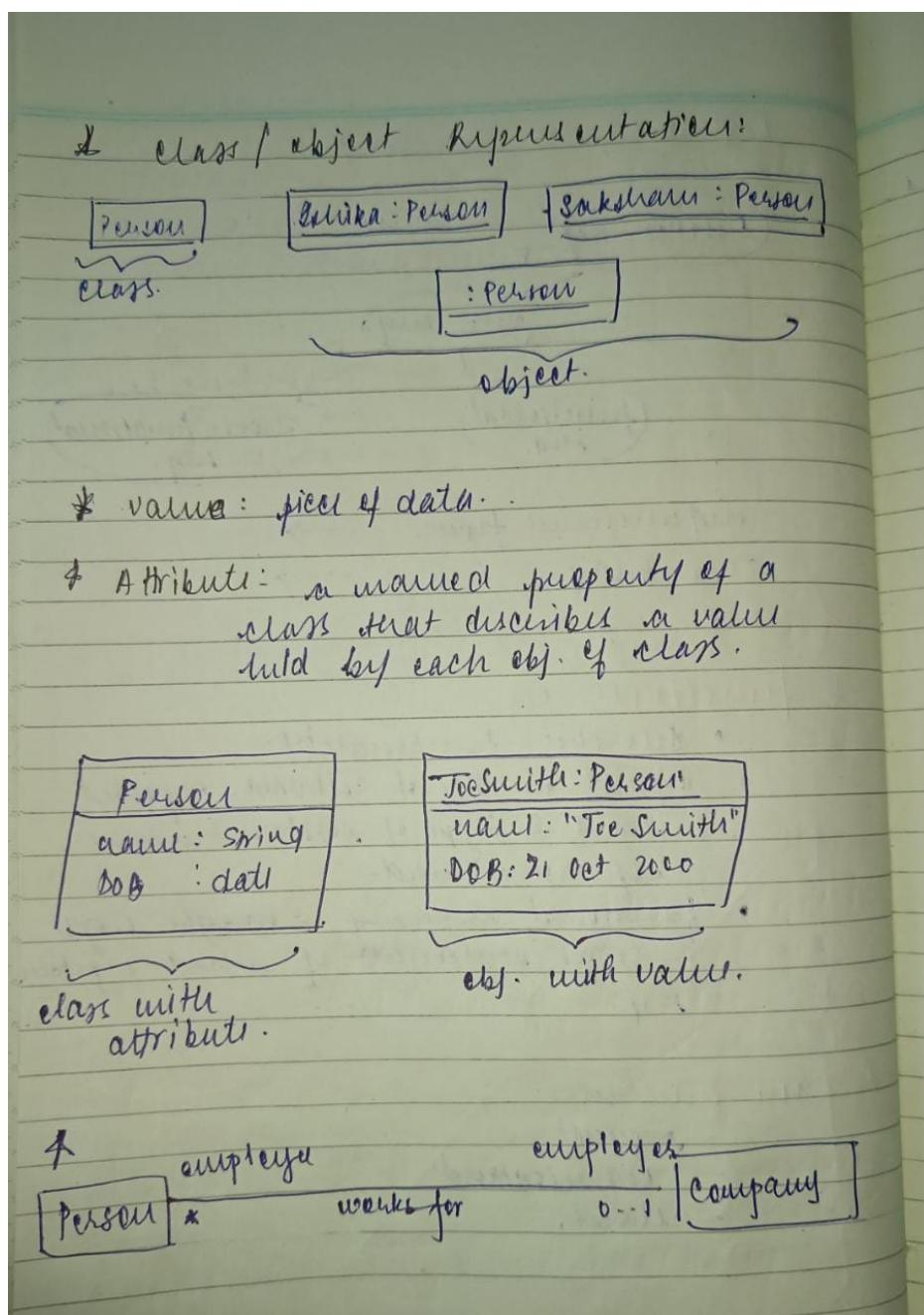


↳ this user case will always be called.





2) CLASS DIAGRAM



* UML class Diagram:

It depicts a static view of an application.

It represents the type of objects existing in the system and the relationship b/w them.

It is one of the most popular UML diagrams.

* Components of a Class Diagram.

consists of 3 parts:-

① upper section
↳ name of class
↳ capital first word

② middle section
↳ constitutes the attributes which describe the quality of class.
↳ attributes are written along their visibility factors

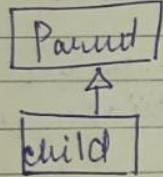
public (+)
private (-)
protected (#)
package (~)

③ Inner section:

- ↳ contains method / operations
- ↳ It demonstrates how a class interacts with others.

Relationships:

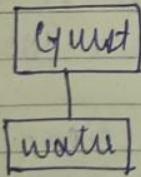
① Generalization:



"is-a"

represent that our statement is based on intuition.

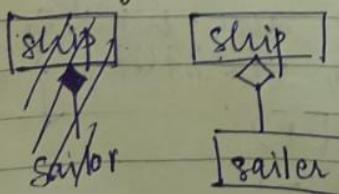
② Association:



"interaction"

used to visualize that two elements interact with / refer to each other.

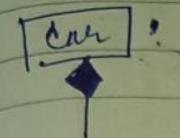
③ Aggregation:



"has-a" denotes ownership
a part-whole relationship.

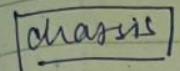
Pants don't die with the wheel.

④ composition:



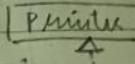
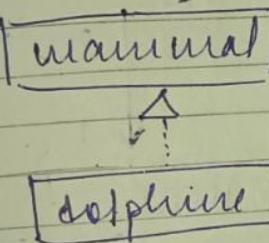
"part-of"

suppose a car
denotes ownership.



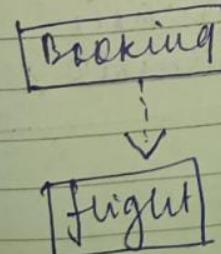
when owning obj. is destroyed, all contained obj. gets destroyed too.

⑤ Realization:



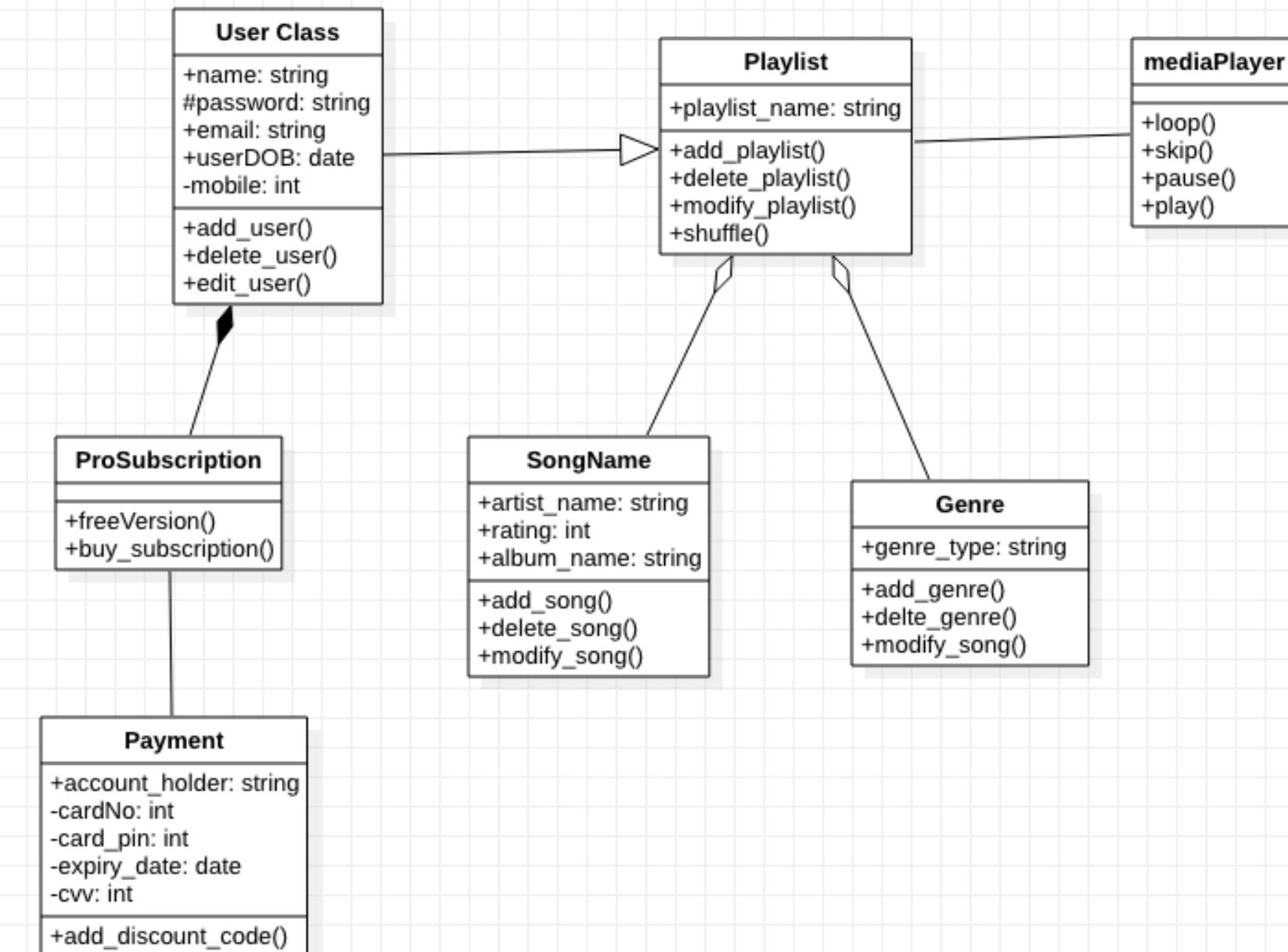
"implements behaviour"
showing that a class implements the behaviour specified by another model element.

⑥ Dependency:



"depends"

indicates that one element depends on another element (e.g. via method parameters)



3) CRC

→ met in UML diagram.

* CRC.

- CRC card = Class Responsibility collaborator card.

- It's a collection of standard index cards that have been divided into 3 parts/sections.

① a class

- ① a class represents a collection of similar objects.

- ② a responsibility is something that a class knows or does.

- ③ collaborator is another class that a class interacts with to fulfill its responsibility.

format of CRC cards:

Class Name:

Responsibility:

what class do/know

Collaborators:

knowing → attributes
Doing → operations.

Customer

Places

knows orders

knows name

knows customer no.

knows order history

order

order

Class : Library.

Responsibilities

knows all available
unreadables

search for unreadables

Collaborators

unreadables

User

knows Name
knows password
knows email
knows user DOB

Login
Delete
edit

Song Name

knows artist name
knows organizing
knows album name

add song
delete song
modify song

ProSubscription

knows card no
knows pin
knows expiry date
knows CVV
knows account holder

make payment
use discount

Playlist

knows playlist name

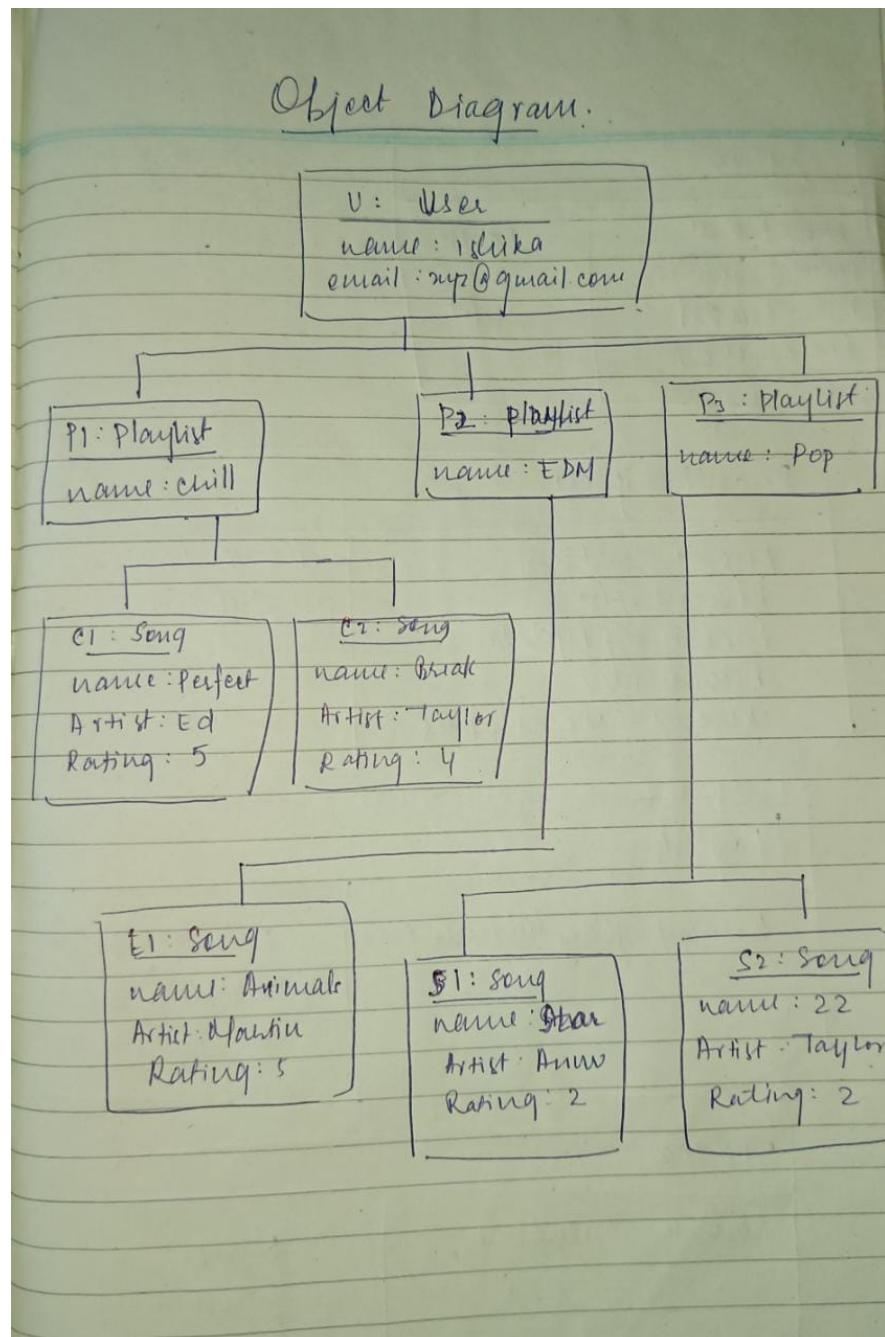
shuffle playlist
sort playlist
modify playlist

media Player

knows song name

play
pause
stop
loop

4) OBJECT NAME



5) SEQUENCE DIAGRAM

* Sequence diagram:
A type of interaction diagram because it describes the how and in what order a group of objects works together.

* Common message symbols:

- ↳ synchronous :
→ used when a sender must wait for a response to a message before it continues.
The diagram must show both the call & reply.

↳ asynchronous:
→ doesn't require a response before the sender continues.
only call should be included in diagram.

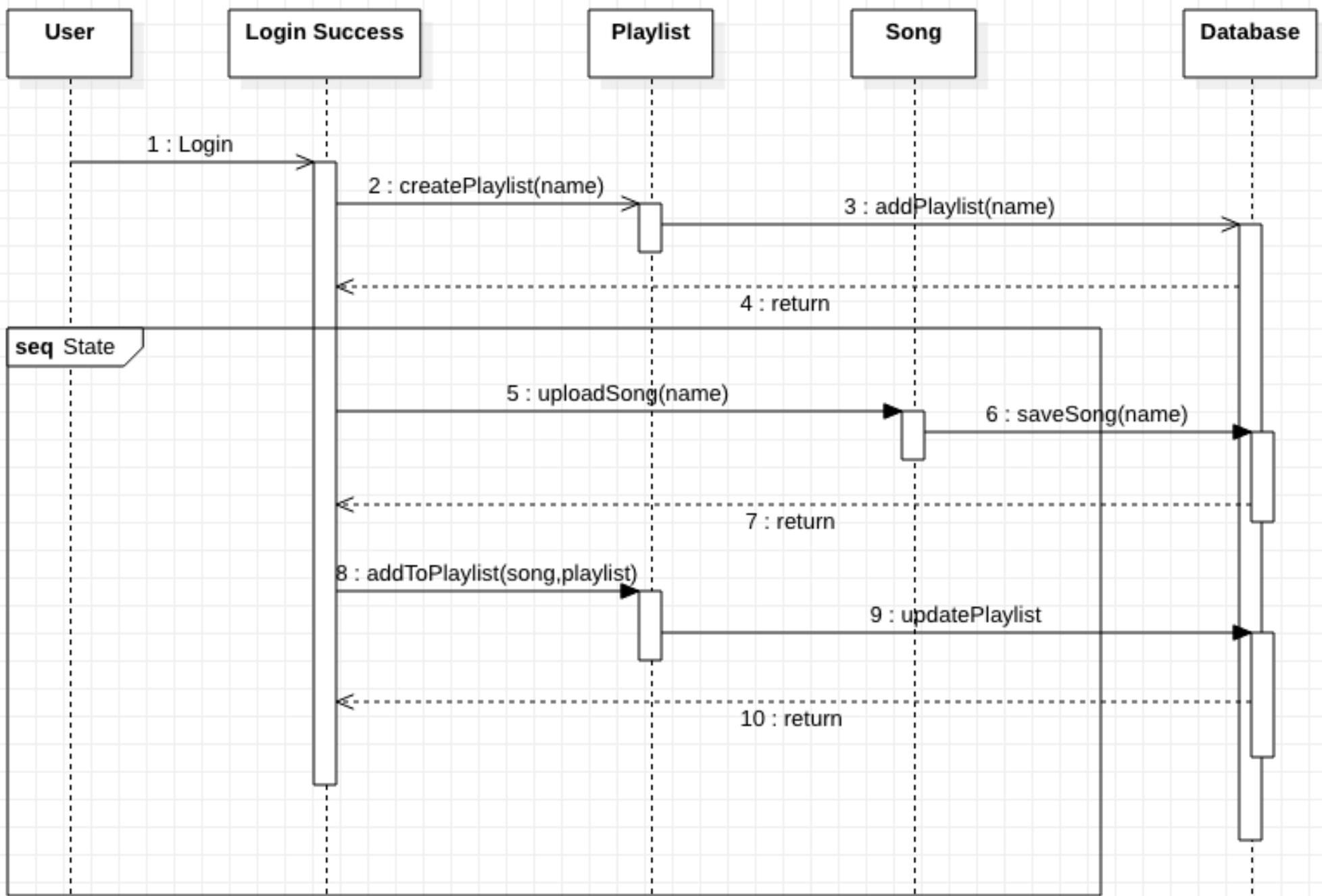
↳ asynchronous return message
← →

↳ asyn. create message:
« create »
— →
This message creates a new object.

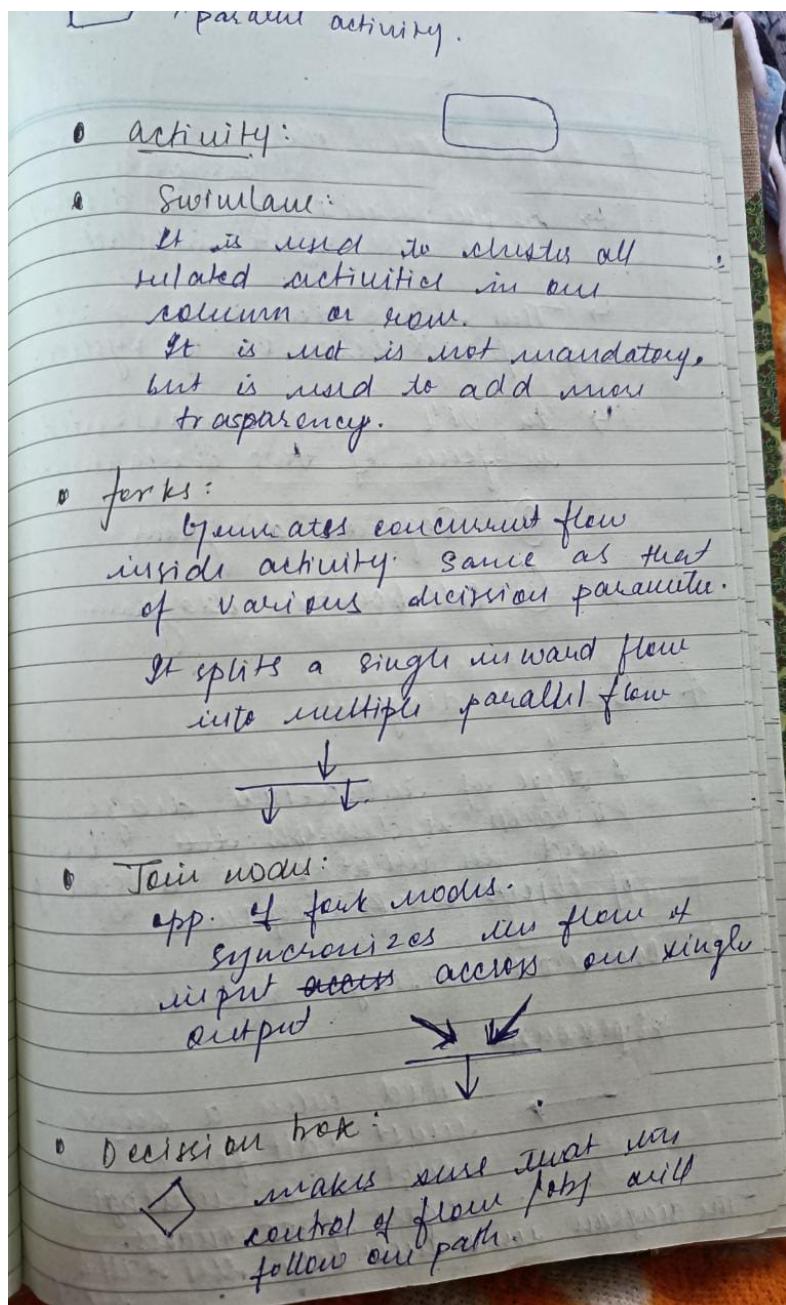
↳ reply message symbol.

A sequence diagram is an interaction diagram that emphasizes the time-ordering of messages.

sd SequenceDiagram1



6) ACTIVITY DIAGRAM



★ Interaction Model: suggests how the objects interact.

- ↳ as the name suggests describes the dynamic behaviours of systems.
- ↳ This interaction is a part of UML it is represented in form of two diagrams.

- ① Sequence diagram
- ② Collaboration diagram.

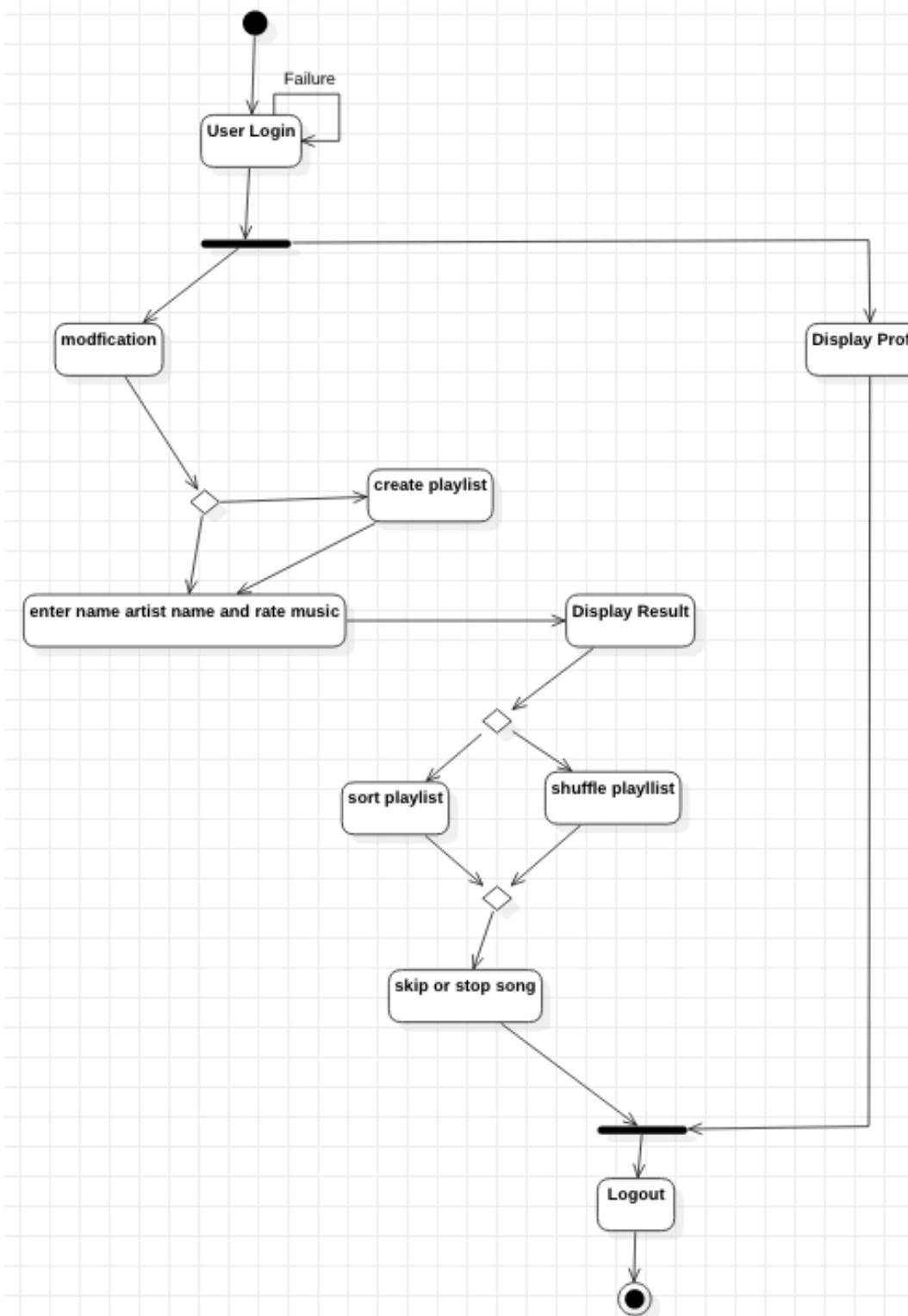
★ Sequence diagram:

A type of interaction diagram because it describes the how and in what order a group of objects works together..

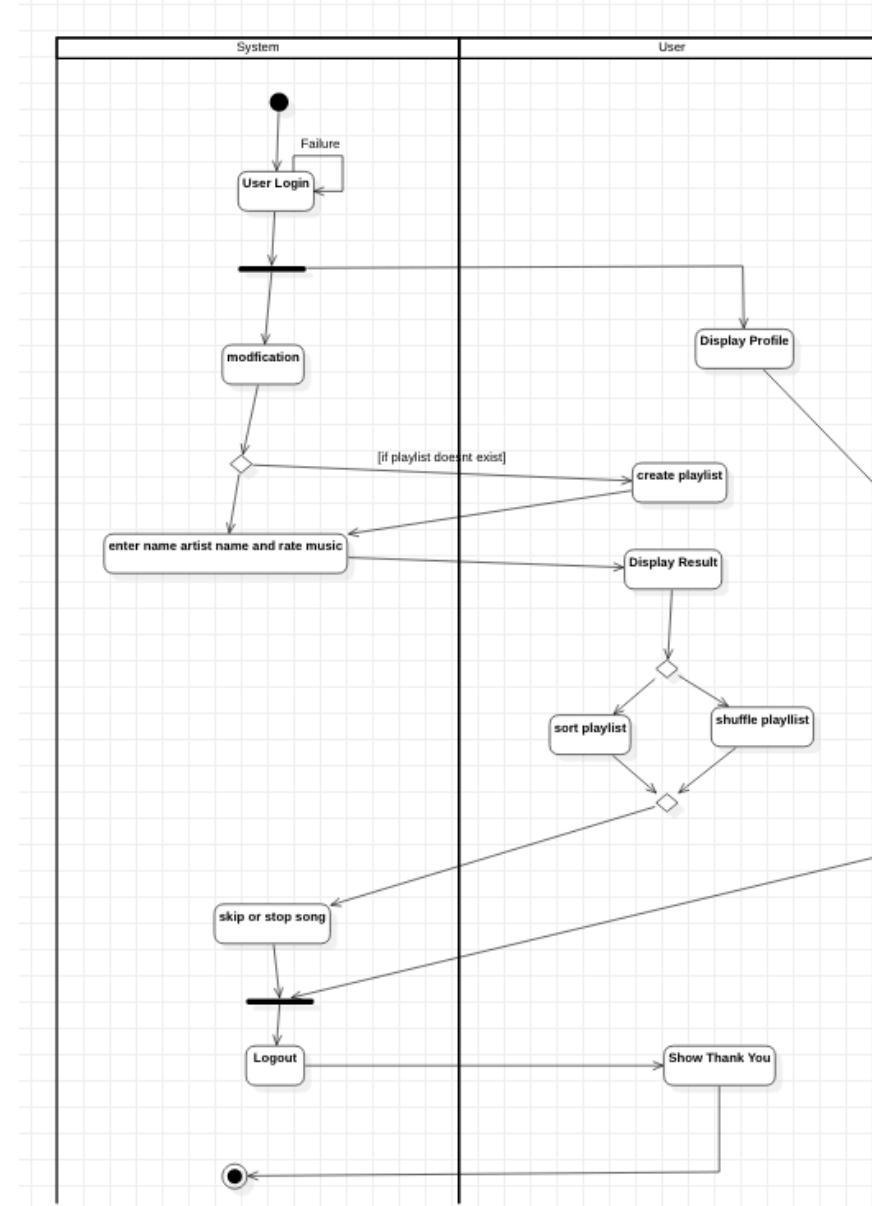
★ Common message symbols:

↳ Synchronous:
 used when a sender must wait for a response to a message before it continues.

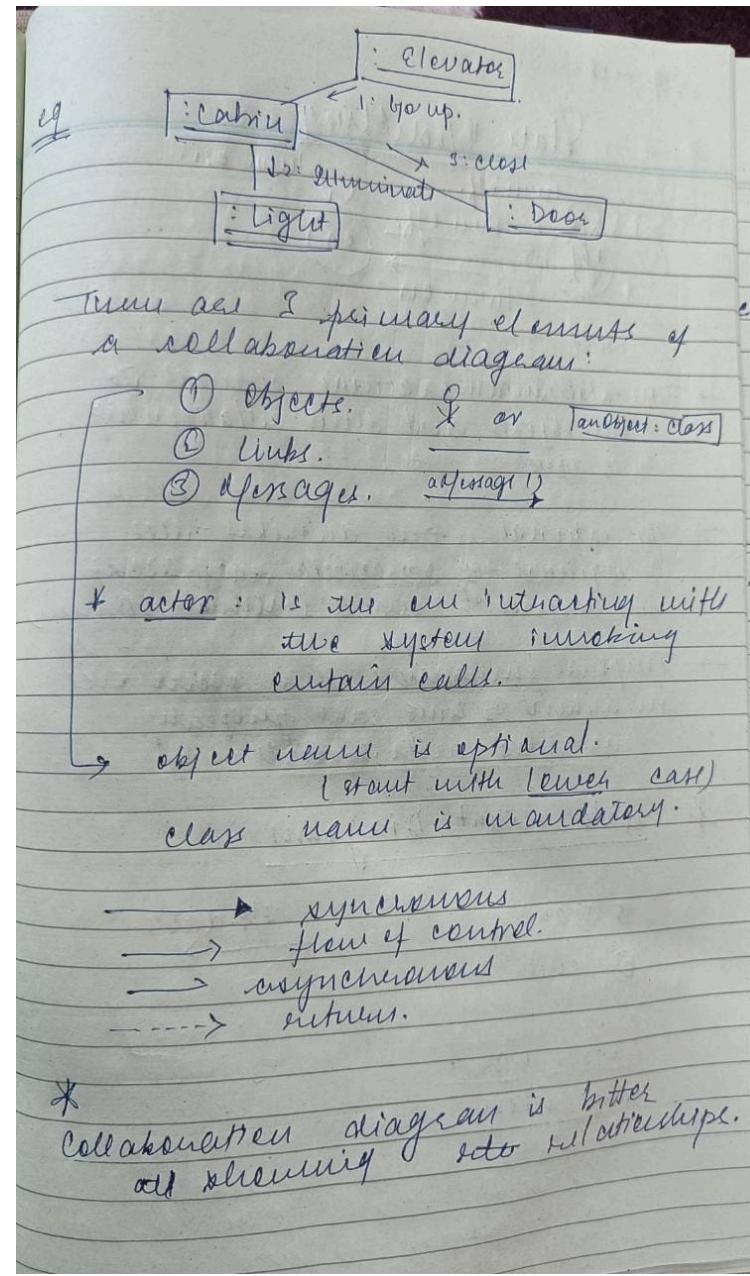
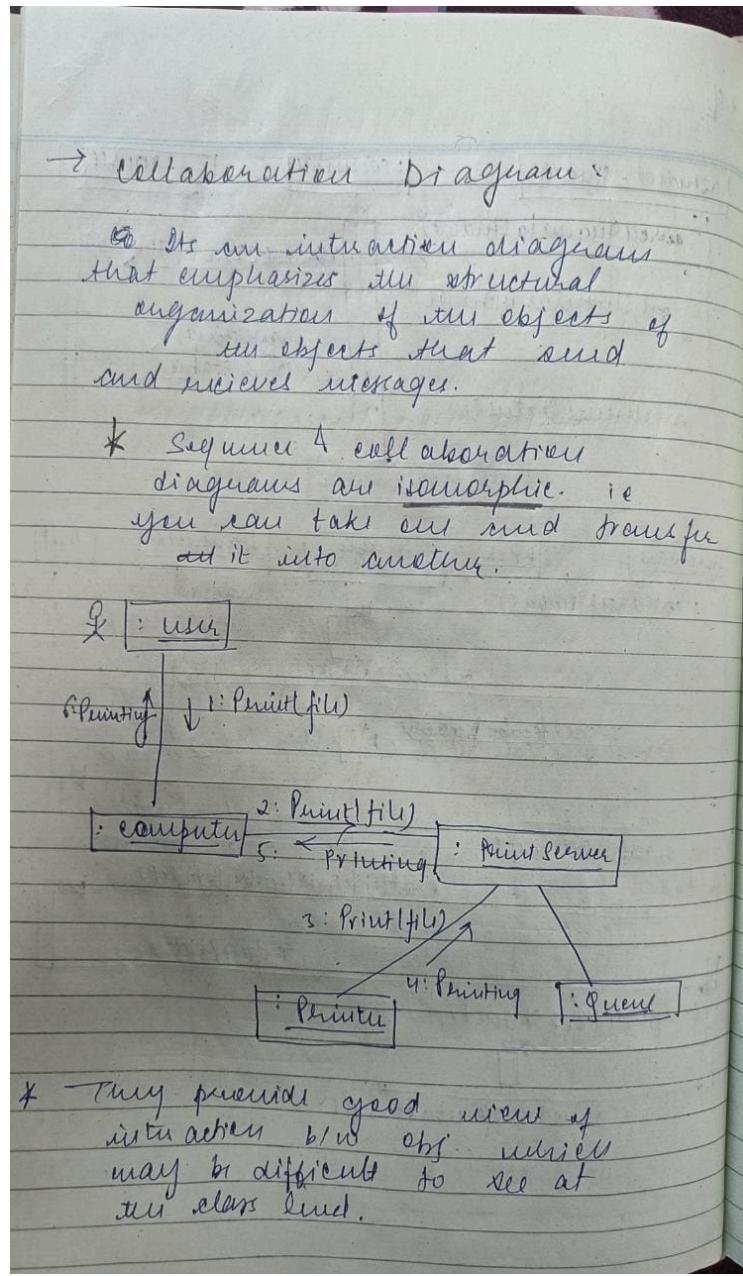
The diagram must show both the call & reply.



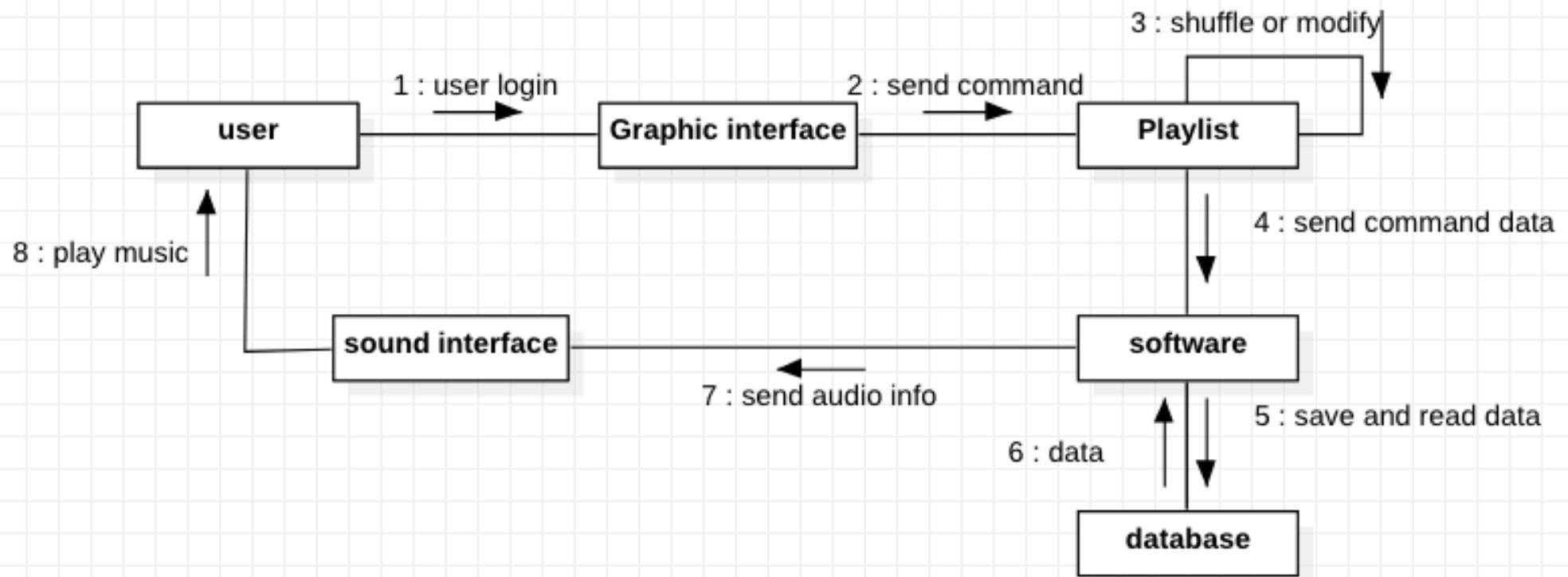
7) SWIMLANE DIAGRAM



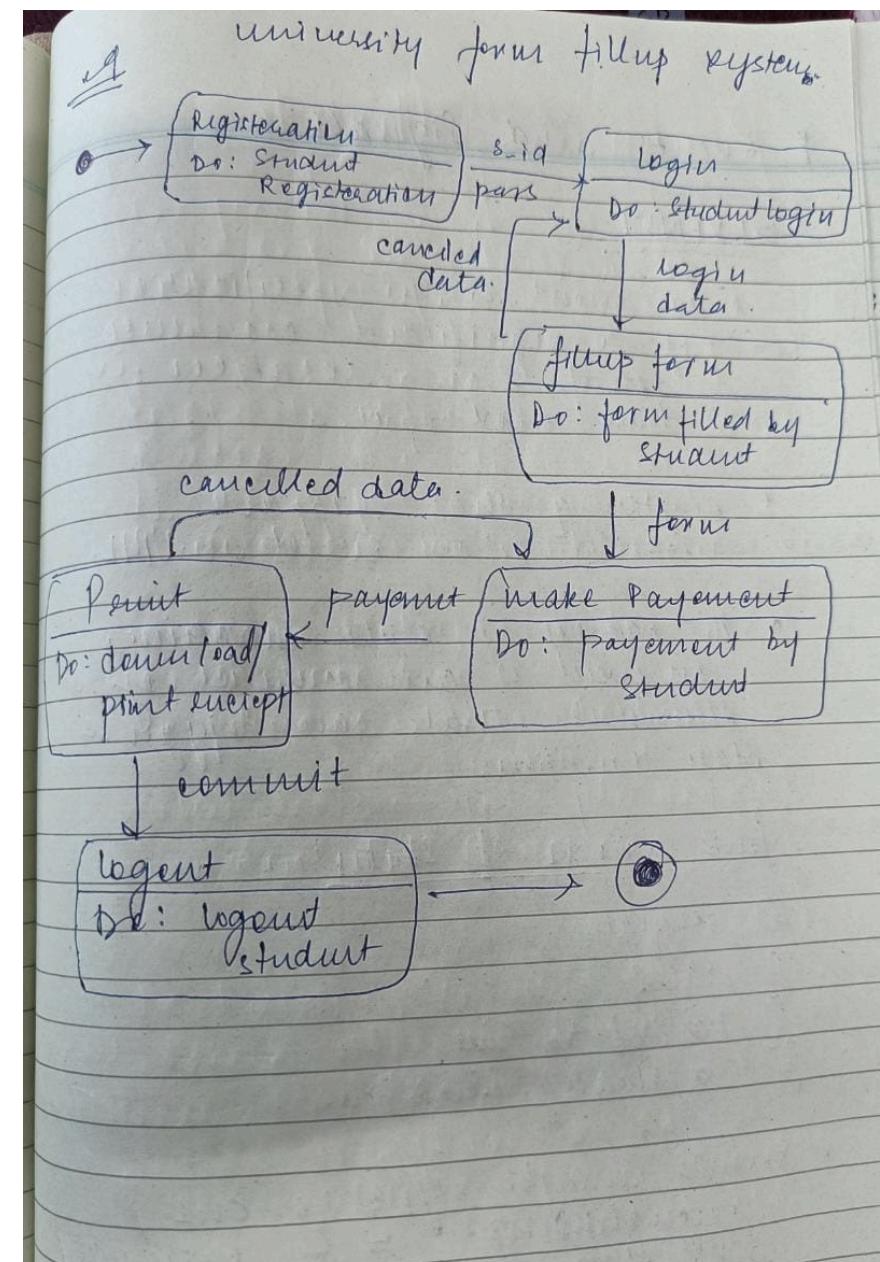
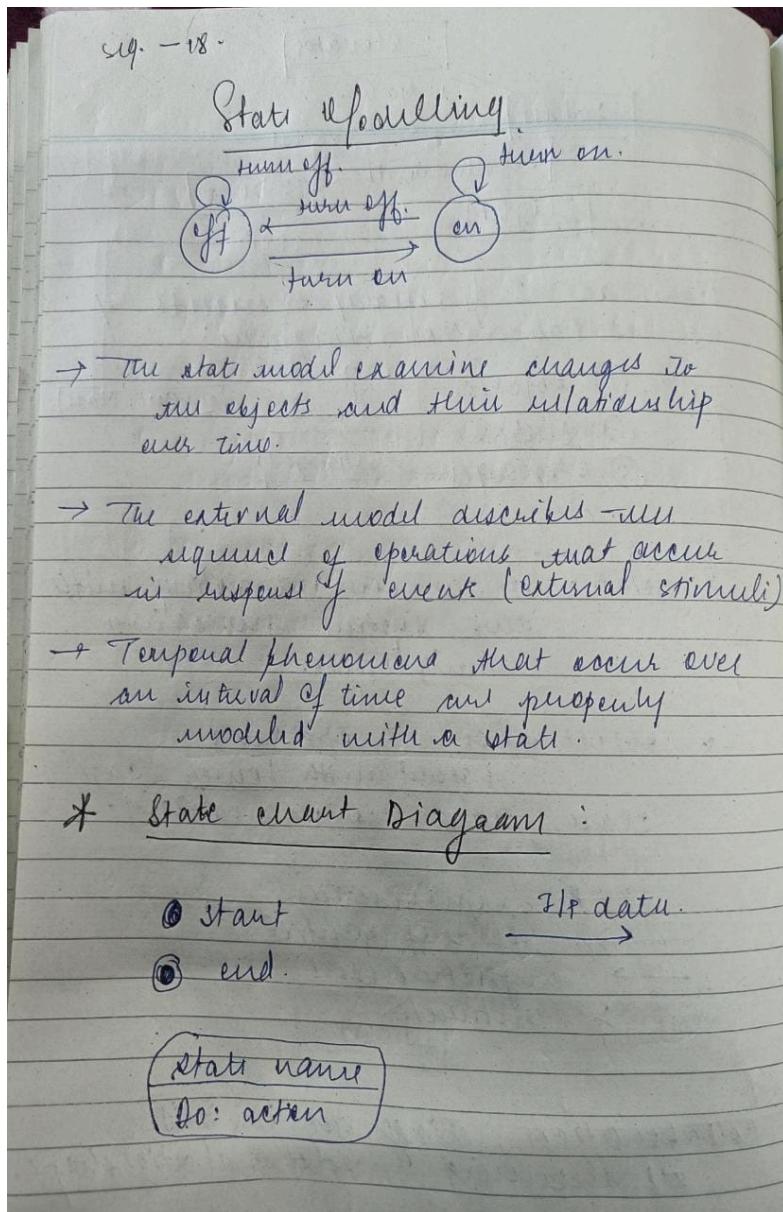
8) COMMUNICATION DIAGRAM

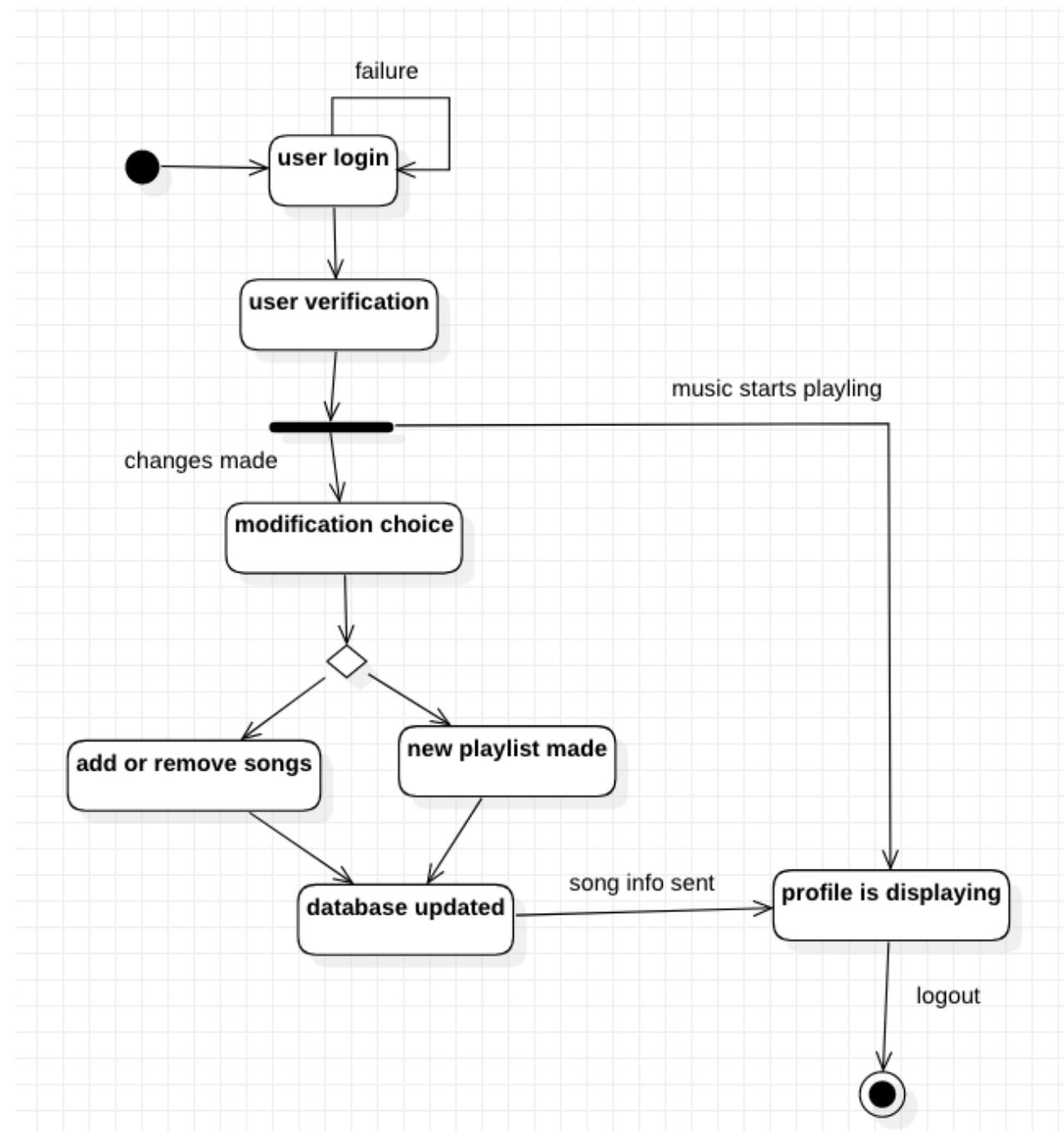


sd CommunicationDiagram1

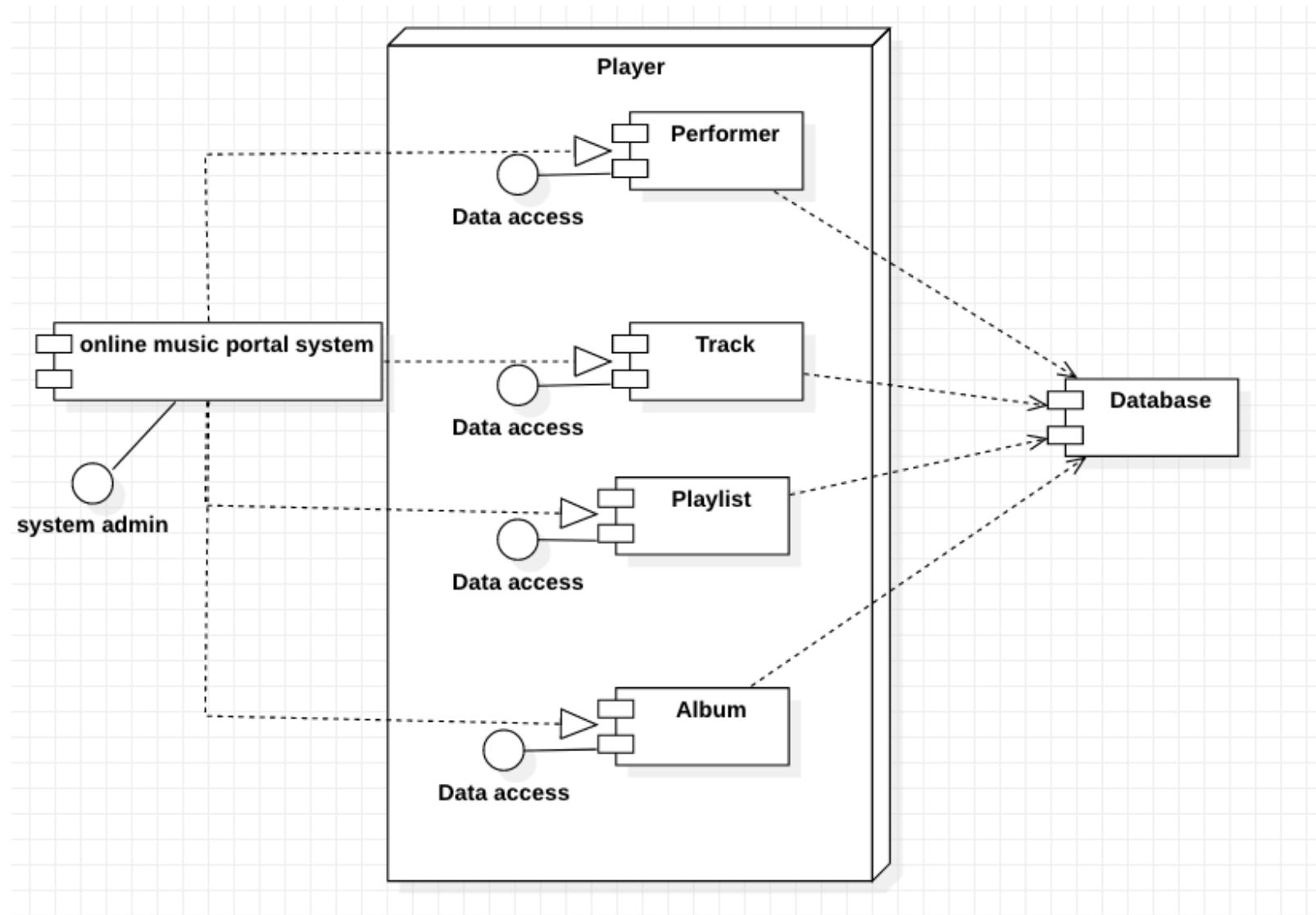


9) STATE DIAGRAM





10) COMPONENT DIAGRAM



11) DEPLOYMENT DIAGRAM

