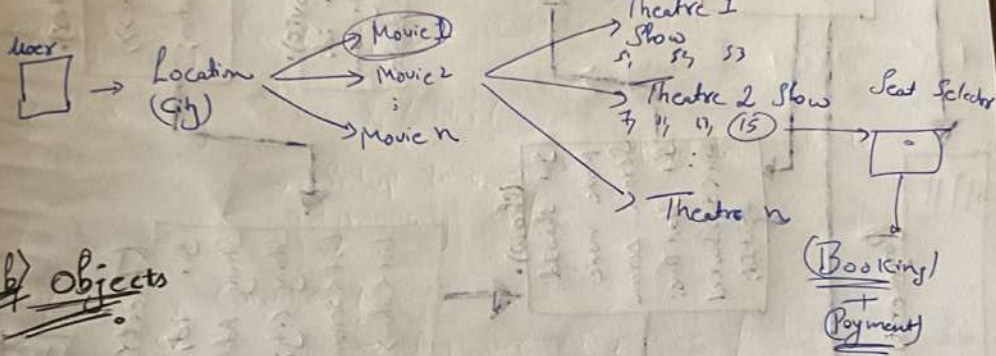


Viii) Movie Ticket (Book My Show)

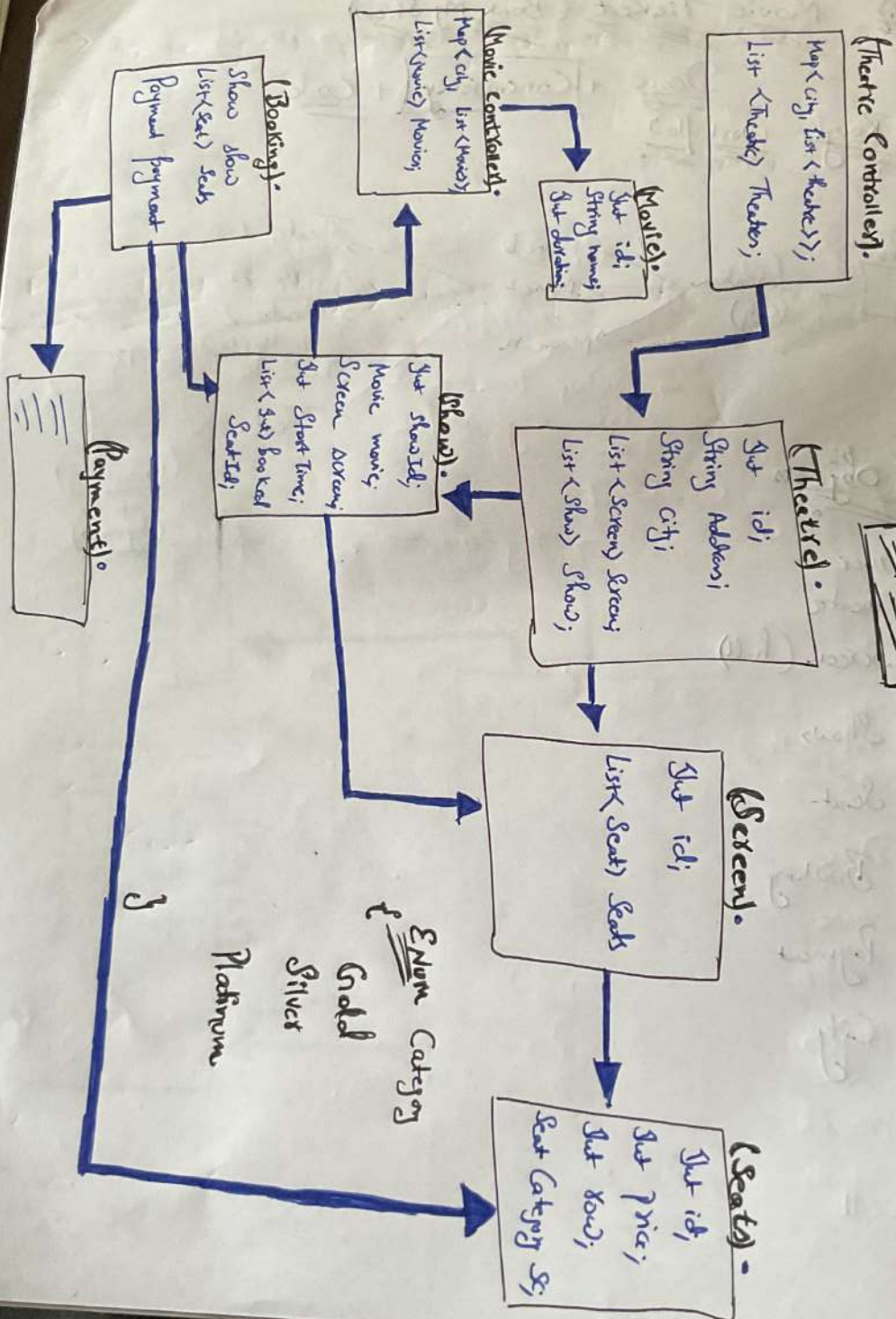
Design + Concurrency + Code

a) Rough workflow



b) Objects

- a) User
- b) Theatre
- c) Screen (hall)
- d) Shows
- e) Seat
- f) Booking
- g) Payment
- h) City



Optim
 i) Persim
 → In the
 current
 → It
 the
 ii) Optim
 → In the
 at
 → when
 the ver
 user
 It is
 writ

iv) Concurrency

Optimistic

Pessimistic

i) Pessimistic Locking

→ In this, we have 2 users, If one of them is reading a current Seat/Element, we will lock it

→ It will be released either when write is done or when the threshold time has been met without doing any write.

ii) Optimistic Locking

→ In this, we have 2 users, both of them can do Read operation at a same time.

→ When, one user wants to update, or write, In this we maintain the versions so like, If during the write, the version that the user has matches with that one of the Seat/Element, that means It is not be updated/written yet so, we allow the Update/Write operation.

- Else, we give message Already occupied or whatever to the user
- If the Lock has not been released, Even After not updating it, we reset the Lock to denlock after some Threshold time.

For Ticket booking ⇒ Optimistic Locking