

Atomicity:

Either the ticket will be created(booked) or it wont, either staff will assigned another shift or he wont

Consistency:

Database remains consistent, there cannot be two tickets for same seat for same date, nor can there be a staff that works more than 40 hours in the week after the transaction

Isolation:

Transactions that may affect each other remain isolated, before inserting, we place a lock on updating and we don't release it till, it either commits or rollbacks

Durability:

After once a ticket has been booked, or a staff has been assigned a shift the change is permanent and saved to hard drive, the changes are not lost

Adding Shift transaction working:

We lock the necessary resource

We try to insert the new one

If the count of shifts the staff has this week exceeds 8 with this(staff working more than 40 hours) then the trigger we wrote cause an abort

Else the shift is assigned and committed

Creating booking for seat:

We lock the necessary attributes

We try to insert a new ticket

If the seat is already booked the trigger for double booking of seats we wrote will cause an abort and roll back

Else the ticket will be created and committed