

## Acid Documentation

**Scenerio 1:**User sends money to another user

If user A wants to send 500 rs to user b.deduct

500 rs from user a walet.add 500 rs to user b

wallet.record the transaction in transaction table

### ACID PROPERTIES MAPPING:

Property	Application
Atomicity	If any step fails ,no money will be transferred
Consistency	Total sum of money before and after transaction will remain same
Durability	Once committed,transaction cannot be lost even after system failure
Isolation	If multiple users send money at same time,transactions cannot intervene with each other

### CONCURRENCY STRATEGY:

Rows of user a and user b should be locked before transaction to prevent conflict.

### **ISOLATION LEVEL:**

repeatable reads means if a transaction reads a row ,it will see the same value if it reads again with same transaction

### **Scenario 2:**

User chooses amount and telecom network .it deducts amount from user's wallet.record recharge in table.

### **ACID PROPERTIES MAPPING**

Property	Application
Atomicity	Either wallet deduction and recharge both happens or nothing happens at all
Consistency	Wallet amount cannot go negative.recharge record must be same as the deducted amount
Durability	Once recharge is done,it is permanently recorded in the database

Isolation	Multiple users recharging at the same time must not intervene with each other
-----------	---

### CONCURRENCY STRATEGY:

Lock the user’s wallet during deduction

### ISOLATION LEVEL:

It should be serializable which will make sure that no lost updates as available,like the same user rehcharges simultaneously.

### Scenario 3:

Users register a debit card ,rs 500 is deducted from user wallet and added to the admin wallet .card is recorded in the admin table

### ACID PROPERTIES MAPPING

Atomicity	If any step fails no fee transferred and card will not be registered
consistency	Total sum of money before and after fee transfer will remain

	same
durability	Once committed, card registration and fee transfer will not be lost even after fee transfer
isolation	If multiple user register card at same time, transactions cannot intervene with each other

### **CONCURRENCY STRATEGY:**

Row of user wallet and admin wallet should be locked before fee transfer to prevent conflict

### **ISOLATION LEVEL:**

Serializable means if two users register cards at same time, each transaction will execute completely and separately so admin wallet is updated correctly without any loss

### **Scenerio 4:**

Merchant based promotion cost to admin.amount is deducted from merchant wallet and added to admin wallet, promtion is

recorded in promotion table and revenue is updated

Atomicity	Either promotion is created or payment is transferred or nothing happens at all
Consistency	Merchant wallet cannot go to negative.revenue table must reflect exact promotion cost paid
Durability	Once promotion payment is committed ,it is permanently recorded in the database,even after system failure
Isolation	Multiple merchants paying for promotions at the same time cannot intervene with each other

### **CONCURRENCY STRATEGY:**

Row of merchant wallet and admin wallet should be locked before transfer to prevent conflicts

### **ISOLATION LEVEL:**

Serializable means if multiple merchants purchase promotions simultaneously revenue table will be updated correctly without any lost updates or phantom read