# Design

## Requirement Analysis

Design and implement program to maintain Equity Positions with unit tests

To manage equity exchange, we decide to develop an application.

* As for transactions, it should include: TransactionID which is Primary key, TradeID which is Trade Identifier, Version which should Starts with 1 , SecurityCode and Quantity.
* Customer can buy or sell those stocks . Every transaction should have unique TransactionID.
* If customer operating on same equity the TradeID should be same.

## Assumptions

* The Positions should be updated after each transaction e.g. once transaction 1 and 2 has arrived, the positions would be REL= +50 and ITC= -40
* INSERT / UPDATE / CANCEL are actions on a Trade (with same trade id but different version)
* INSERT will always be 1st version of a Trade, CANCEL will always be last version of Trade.
* For UPDATE, SecurityCode or Quantity or Buy/Sell can change
* For CANCEL, any changes in SecurityCode or Quantity or Buy/Sell may change and should be ignored
* The transactions can arrive in any sequence

## Approach

|  |  |  |
| --- | --- | --- |
| Class Name | Method | Remarks |
| Transactions.java | Getter/setter | It’s a Entity Class |
| Service.java | Sell()  Buy() | Sell method is processing logic for customer reduce quantity belongs to him.  Buy method is for increase equity quantity |

## Test cases

Test method put in main class to test the logic in service.