

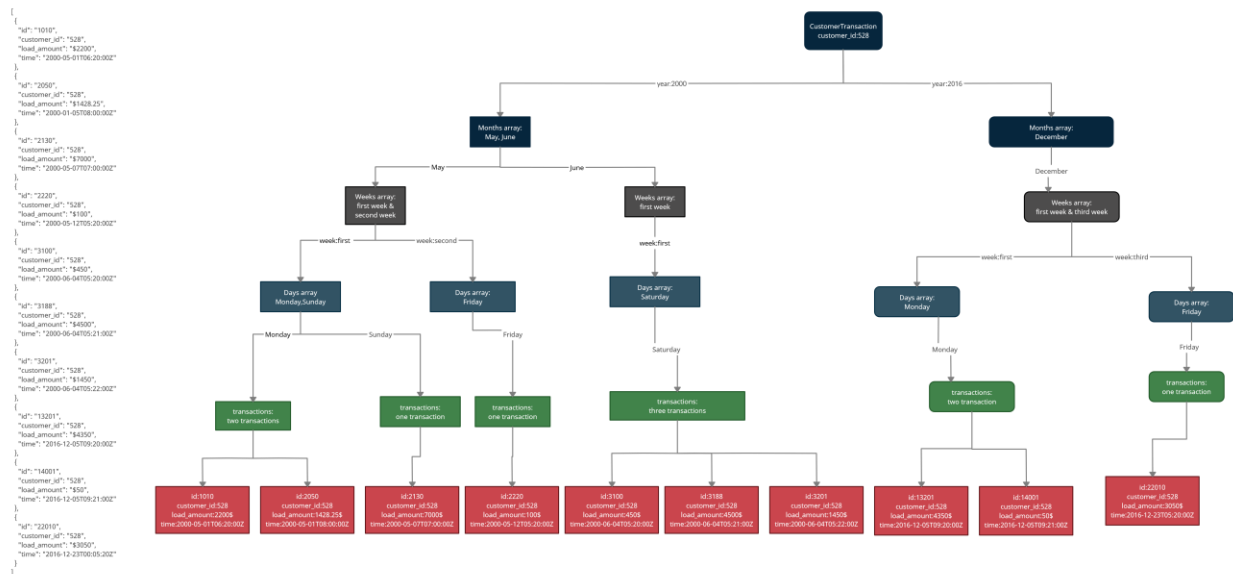
My solution step by step :

1.Step one: read all transaction. e.g.

```
{
  {
    "id": "1010",
    "customer_id": "528",
    "load_amount": "$2200",
    "time": "2000-05-01T06:20:00Z"
  },
  {
    "id": "2050",
    "customer_id": "528",
    "load_amount": "$1428.25",
    "time": "2000-01-05T08:00:00Z"
  },
  {
    "id": "2130",
    "customer_id": "528",
    "load_amount": "$7000",
    "time": "2000-05-07T07:00:00Z"
  },
  {
    "id": "2220",
    "customer_id": "528",
    "load_amount": "$100",
    "time": "2000-05-12T05:20:00Z"
  },
  {
    "id": "3100",
    "customer_id": "528",
    "load_amount": "$450",
    "time": "2000-06-04T05:20:00Z"
  },
  {
    "id": "3188",
    "customer_id": "528",
    "load_amount": "$4500",
    "time": "2000-06-04T05:21:00Z"
  },
  {
    "id": "3201",
    "customer_id": "528",
    "load_amount": "$1450",
    "time": "2000-06-04T05:22:00Z"
  },
}
```

```
{
  "id": "13201",
  "customer_id": "528",
  "load_amount": "$4350",
  "time": "2016-12-05T09:20:00Z"
},
{
  "id": "14001",
  "customer_id": "528",
  "load_amount": "$50",
  "time": "2016-12-05T09:21:00Z"
},
{
  "id": "22010",
  "customer_id": "528",
  "load_amount": "$3050",
  "time": "2016-12-23T00:05:20Z"
}
{
  "id": "52010",
  "customer_id": "876",
  "load_amount": "$250",
  "time": "2017-10-23T00:05:20Z"
},
{
  ...
}]
```

2.Step two: create a tree per customer base on above transactions. e.g.



3.Step three: evaluate input customer transactions (is accepted or not) base on the rules (velocity limits).

4.Last step: print all transactions with status that shows every transaction is accepted or not.

Runs commands:

1. run test: Jest framework : npx jest / npx jest --coverage
2. compile typescripts to js: tsc app.ts --watch
3. run app: app.js run to show output (transactionResponse) : nodemon app.js

RUNS	test/handler/tree_node/transaction/nodeHandler/transaction_node_handler.test.ts
PASS	test/validation/transactin_nodevalidator.test.ts
PASS	test/handler/transaction_handler/transaction.test.ts
PASS	test/evaluator/transactionRequest/evaluator.test.ts

File	% Stmts	% Branch	% Funcs	% Lines	Uncovered
All files	95.36	90	85.56	95.8	
evaluator/transactionRequest	100	100	90.91	100	
evaluator.js	100	100	90.91	100	
handler/transactionHandler	100	100	92.86	100	
transaction.js	100	100	92.86	100	
handler/tree_node/transaction/inquiry	100	100	91.67	100	
inquiryCustomerTransaction.js	100	100	91.67	100	
handler/tree_node/transaction/nodeHandler	72.41	100	50	72.41	
TransactionNodeHandler.js	72.41	100	50	72.41	14-15, 18
handler/tree_node/transaction/tree	89.29	75	80	88.89	
customerTransactionHandler.js	89.29	75	80	88.89	20, 22, 26
models/common	100	100	100	100	
dateTimeParts.js	100	100	100	100	
models/transaction	87.88	57.14	83.33	93.55	
transaction.js	100	100	100	100	
transactionRequest.js	80	57.14	75	88.89	5-6
transactionResponse.js	100	100	100	100	
models/tree-transaction_viewmodel	100	100	100	100	
customerTransaction.js	100	100	100	100	
dailyTransaction.js	100	100	100	100	
monthlyTransaction.js	100	100	100	100	
weeklyTransaction.js	100	100	100	100	
yearlyTransaction.js	100	100	100	100	
setting/transaction-error	100	100	100	100	
errors.js	100	100	100	100	
setting/velocity	100	100	100	100	
limits_config.js	100	100	100	100	
utility	95.24	50	66.67	95.24	
dateTimeUtility.js	95.24	50	66.67	95.24	12
validation	100	100	84.62	100	
transactin_nodevalidator.js	100	100	90	100	
validatorDataSrc.js	100	100	66.67	100	

Test Suites: 7 passed, 7 total
Tests: 45 passed, 45 total
Snapshots: 0 total