**ATM LLD Practice**

Class AtmManagement (Performing operations)

— Carduser (constructor) D

— card\_to\_account\_map (constructor) D

—- Authenticate the User and process the details D  
 def authenticate(self,User):

Pass

—- Get accountnumber from card: D

Def get\_account\_number\_from\_card(self): // card is for

Return self.card\_to\_account\_map[cardUser.getCardNumber()]  
 —-Balance Details from the account

Class ATMController  
 — AccountManager  
 — card\_to\_user\_map  
 —card\_to\_account\_map  
 –Process card(card,transaction\_type,amount) // Based on authentication

Class AccountManager: // Take care of all the accounts and manage them

Dict = {}

add(account):

Self.dict[account\_number] = account

get\_all\_accounts()

Return dict

get\_account()

Return dict.get(account\_number)

Class CardUser (Details of user) D  
 — PIN  
 — Card Number

— Name

// getters for card number and Name

Class Account: Thread safe  
 — account\_number (constructor)  
 — balance (constructor)

–getaccountNumber()

—getbalance()

—deduct\_balance(amount) — called by withdraw method

—credit\_balance(amount) - called by deposit method

– str class to display the account details

Class Transaction: Thread safe (update the details in the account

Class TransactionTpe(Enum):

Withdraw = 1

Deposit = 2

-account  
 -amount

Transaction\_type = TransactionType.withdraw

Def process(self):

if(self.transaction\_type == TransactionType.withdraw):

– call withdraw method // private

Elif(self.transaction\_type == TransactionType.deposit):

- Call deposit method // private

transactionid = generateRandom(transactiontype)   
 Return transactionId

withdraw():

If account.getbalance()>=amount:

deduct\_balance(amount)

Deposit:

credit\_balance(amount)

generateRandom(type):// Private class

Class cash\_dispenser Thread safe (handle the dispense of the cash)  
 — account number

–amount

dispense\_cash():

print(f”cash withdrawn is ${self.amount} from account ${self.account\_number}

Class ATMDemo: // Will do all the account data management and Card data management   
 Will contain the card data which is hardcoded or input by the user of the system

Will call process\_card method once card details are added and transaction type is also selected along with amount entered