**Requirements**: Create design (class diagram and schema design) and write working code for a **command-line payment** application that supports the following use cases.

In future:

* a user might create via email ID/ google login; fb logic
* I might want to refund to original payment source
* How to

Behaviour:

* Check for balance before making payment via wallet
* GUI + APIs

1. Allow users to create accounts via their phone number and password.

| RegisterUser [phone\_number] [password] |
| --- |

1. Allow users to update their profile details.

| UpdateUser [user\_id] [name] [email] [phone\_number] |
| --- |

1. Send money to another user of the application. (Send money to another phone number)

| CreateTransaction PAYTM [from\_user\_id] [to\_user\_id] [amount] |
| --- |

1. Send money to a bank account.

| CreateTransaction BANK [from\_user\_id] [account\_number] [ifsc\_code] [amount] |
| --- |

1. Allow users to make payment for the transaction via Card/ UPI/ Netbanking

| MakePayment [transaction\_id] [payment\_method] [... details related to payment method ...] |
| --- |

1. Allow users to refund a transaction. The refund amount will go to the original source.

| RefundTransaction [transaction\_id] |
| --- |

1. Allow users to view transaction history.

| ViewTransactionsHistory [user\_id] |
| --- |

Amazon:

1. Create an order -> OrderID -> In paytm, transaction
2. Make payment for that order -> PaymentReferenceNumber

This will allow me to retry payment for a transaction

Deepti -> kartiki -> 1000 => Transaction

* Debit Card => Payment
* Payment Processing
* Retry
* Credit Card => Payment
* Success

Transaction

* List<Payment>