**E-Wallet Application PRD**

Features/Functionalities:

* User of E-Wallet can transfer Money to Some other User.
* User will have their account with current balance which will be incremented/decremented based on transactions
* User should be able to see recent n transactions.
* Whenever a user transacts, they should be getting email notification about their transaction.
* User should be able to update/delete their account.
* Whenever User registers, He/she should get the welcome amount in wallet.(Communication between User and Wallet using Kafka).
* Whenever there is any Transaction, Notification Service should send the Mail to User(For Sms - 3rd Party Integration required)
* Once Transaction is Successful, Publish Kafka Message which will be consumed by multiple services. Eg- wallet service will consume and update wallet balance.
* Once Transaction is Successful both Sender and Receiver should be notified and Only Sender in case of Failed Transaction.

Application Design:

* E-Wallet will be Parent Application Containing Various Microservices (E.g. - User Service, Transaction Service, Notification Service, Wallet Services, etc) with each service having its own database.
* Application Should follow Microservice Architecture. Communication between services should be done using Kafka.

Methods:

* UserController:
  + @PostMapping("/user") public void createUser(@RequestBody UserCreateRequest userCreateRequest)
  + @GetMapping("/user") public User getUserDetails()
  + @GetMapping("/admin/all/users") public List<User> getAllUserDetails()
  + @GetMapping("/admin/user/{userId}") public User getUserDetails(@PathVariable("userId") String userId)
* TransactionController:
  + @PostMapping("/txn") public String initiateTxn(@RequestParam("receiver") String receiver, @RequestParam("purpose") String purpose, @RequestParam("amount") Double amount