**CORE USER REQUIREMENTS**

**NON-FUNCTIONAL USER REQUIREMENTS (PERFORMANCE OR USABILITY):**

* **Transaction Security:**
* Money to transfer should not be greater than the account balance.
* Debit transactions should only be from chequing account.
* Do not transfer money if account if empty.
* Transaction pin must be confirmed.
* Simulated APIs for different banks to keep transaction secures
* **Account Management:**
* After every transaction, sent or received, the account balance must be updated.
* Transaction history must be recorded
* validate pin to view transaction history
* Separate Simulated API for account operations

**FUNTIONAL USER REQUIREMENTS (PROCESS OR DATA):**

* **Create Online Account:**
* Checks if account number exists
* Validates email, username and password inputs.
* **Login Online Account:**
* Verify if online account(email/username) exists and password is correct.
* **View account balance:**
* Always shows current account balance on login.
* Shows updated account balance when refresh button is clicked
* **Send/Transfer money:**
* Recipient account number is confirmed.
* Customer chooses which bank (internal or external).(APIs)
* Verify if customers account is active and proceed with transaction.
* Customer confirms transaction with pin.
* **Update Account info:**
* Can only update phone, email, and password.
* To change phone, email or password, must be verified with old password.
* **Receive money:**
* Update account balance on receipt
* Record deposit transaction
* **Record Transaction and View Transaction history:**
* Details must include sender’s name, last 3 digits of senders account number, amount, bank name, date and time of transaction, status (only successful)
* Details must include receiver’s name, full account number, amount, bank name, date and time of transaction, status (failed or successful)
* View transaction history button shows transaction done in the last 7 days on default.

**NON-CORE USER REQUIREMENTS**

**NON-FUNCTIONAL USER REQUIREMENTS (PERFORMANCE OR USABILITY)**

* **Application security:**
* Only accept valid inputs
* **Account Security:**
* Passwords are encrypted when stored in the database and encryption keys are saved on the local device storage an

**FUNTIONAL USER REQUIREMENTS (PROCESS OR DATA)**

* **Create Account:**
* Email, username and password input fields are mandatory
* **View account balance:**
* Shows updated account balance when refresh button is clicked
* **Send/Transfer money:**
* Customer name is displayed after account number is verified
* Can transfer from chequing to savings account and vice versa
* **Update Account info:**
* Rejects if new email or password is the same with old one.
* **Receive money:**
* Give a notification on deposit.
* **Record Transaction and View transaction history:**
* Details must include sender’s name, last 3 digits of senders account number, amount, bank name, date and time of transaction, status (failed or successful)
* A trigger to export Account statement for the month.
* Option to see transaction history for a specified period of time.

**HOW TO DETERMINE USER REQUIREMENTS.**

**Root cause analysis**

**Duration analysis**

**Informal benchmarking**

**Outcome analysis**

**Technology analysis**

**Activity elimination**