Frequently Asked Questions(FAQs)

On Programming Assignment 2 (Term Project)

Question 1: Is it sufficient to use only one device for the entire implementation, or are three separate devices mandatory?
Ans: It is clearly stated in the assignment that you must use ***three*** separate laptops for the 3 different entities.

Question 2: How should the 3 laptops communicate with each other?
Ans: There should be 3 laptops which will send information to each other via LAN using ***Socket Programming*** .

Question 3: Which data structure can we use ?
Ans: You can use Data structures like vector<struct> in cpp, or list in python to maintain records of users, banks, and merchants.

Question 4: How to generate and use the QR code ?
Ans: There are libraries for making QR code, you can use that. You can scan the QR using your ***mobile phones*** and manually enter the displayed data in the laptop.

Question 5: It is mentioned that the Merchant ID and UID must be a 16-bit hex number. However, the output of the SHA-256 algorithm would be a 64-digit hex number. Is there any specific conversion algorithm to reduce the size to 16 digits?
Ans: Yes, SHA-256 generates a 64-character (256-bit) hex output, but to reduce it to 16 hex digits (64 bits), you can use the following method: ***Truncation*** – Take the first 16 characters or the last 16 characters of the SHA-256 hash.

Question 6: Does the MMID also need to be created through the SHA256 algorithm or should it be a simple concatenation ?
Ans: Yes, the MMID must be generated using the ***SHA-256*** algorithm.

Question 7: Are we allowed to use in-built ciphers for encryption and decryption, or do we need to implement them from scratch?
Ans: You can use standard built-in ciphers.

Question 8: Should we use an external database (such as SQL or MongoDB) to store merchant and user information, or is an in-memory solution sufficient?
Ans: The assignment does not mandate using an external database. However, you must store bank details, as the bank maintains a centralized blockchain ledger and handles user and merchant registrations. While implementing, you can use data structures like vector<struct> in C++ or list in Python to maintain records of users, banks, and merchants.

Question 9: Should the QR scanning functionality be fully implemented, or is a conceptual demonstration acceptable?
Ans: QR scanning must be fully implemented (**as mentioned in the document**). This includes generating a QR code containing the encrypted Merchant ID (MID) using LWC and ensuring that the user can scan and decode it to retrieve transaction details.

Question 10: What is VMID ?
Ans: VMID (Virtual Merchant ID) is generated when a merchant enters their Merchant ID (MID) into the UPI machine. The UPI machine encrypts the MID using Lightweight Cryptography (**LWC**) to produce the VMID. This VMID is embedded in the QR code, which the user scans to initiate a transaction. Please ***refer section 3.1*** in the assignment for more details.

Question 11: Are we required to develop a front end for the application?
Ans: The document does not explicitly require a frontend. The primary focus is on the backend implementation of encryption, blockchain, and transaction handling. A command-line or basic graphical interface would be helpful for testing. ***While not mandatory, a front end could help in visualizing the workflow more effectively.***

NOTE: For the demo, you must ensure that all necessary data (bank details, merchants, users, transactions) is present in your database to demonstrate a fully functional system.

For any further clarification regarding the assignment, the students may approach the TA of the course
NISHCHAY DEEP (f20213144@hyderabad.bits-pilani.ac.in)