Meet 1 : Offline

SECURE QR CODE

Section A: Create Basic Functionality

Step 1 : Create form on respective frontend technology.
Form contains name, gender, age, phone number and profile picture.
Step 2 : Take the data from form and convert it into QR code.
Step 3 : Scan the QR code Retrieve the data from QR code.

Section B: Add a cryptographic method.

Step 4: For encryption and Decryption, we use the AES algorithm (Advanced Encryption Standard is a widely used symmetric encryption algorithm, known for its security, efficiency, and flexibility in safeguarding data with key lengths of 128, 192, and 256 bits.)

Section C : For Digital Signature

Step 5 : Process for digital Signature:

1. **Key Pair Generation**: Generate a private-public key pair.
2. **Data to Be Signed**: Select the data you want to sign.
3. **Hashing**: Create a hash of the data.
4. **Signing**: Generate a digital signature with the private key.
5. **Attach Signature**: Combine the signature with the data.
6. **Verification**: Recipients compute a hash of the received data.
7. **Decryption and Comparison**: Decrypt the signature and compare hashes.
8. **Signature Validation**: Verify if the hashes match for signature validity.
9. **Secure Key Management**: Safeguard the private key for security.

Meet 2: Online(Google Meet)
Meeting Report:

Date: 06/09/2023
Time: 3:00 - 3:30
Location: Google Meet

Agenda:

The primary objective of this meeting was to discuss and understand the tech stack and the overall process flow for developing an Android application using Flutter. Additionally, we deliberated on encryption and decryption methods, digital signature algorithms, and the image compression component of the project.

Meeting Summary:

1. Tech Stack & Process Flow:
   - We have collectively decided to develop the Android application using Flutter. This choice will allow us to create a cross-platform app efficiently.
   - The project's overall process flow was discussed briefly to ensure everyone is on the same page.

2. Encryption and Decryption:
   - It was decided that for encryption and decryption, we will implement the Chacha20-poly algorithm. This choice was made considering its security and efficiency.

3. Digital Signature:
   - The group consensus is to use the ECC (Elliptic Curve Cryptography) algorithm for digital signatures. ECC provides a good balance between security and performance.

4. Data Storage:
   - The application's data storage approach needs to account for the limitation of storing text data and images within a QR code, which can hold a maximum of 3 KB of data. To accommodate this constraint, we will implement image compression techniques to efficiently encode images into QR codes while ensuring optimal storage usage.

5. Image Compression:
   - An image compression component will be integrated into the application. The meeting attendees agreed to conduct thorough testing of various image compression and decompression algorithms using a diverse set of images.
   - A detailed report on the results of these tests will be generated to determine the most suitable image compression algorithm for our application.

Assigned Tasks:

- Each team member will research and test different image compression and decompression algorithms.
- The results of these tests will be documented in a comprehensive report, including factors such as image quality.

Member 1 : Himanshu Upadhyay
Member 2 : Aditya Waskar
Title : Offline e-Pramaan
College Name : Shree LR Tiwari College Of Engineering