

Classes and Relations

There are 7 classes in the project:

- 1) Two main entity classes: **Bank.java** and **Client.java** represent the two communicating entities using RSA digital signature. They each have methods for encrypt/decrypt message.
- 2) Three data structure classes: **ClientAccount.java**, **EncryptedMessage.java**, **TransactionRecord.java** are data structures class that are mostly used for medium for transferring info between entities and/or logging information.
- 3) One main simulation class: **SecureBankVerificationSimulator.java** is the main class and walk through the 3 steps of RSA encryption simulation
 - **Step1. Generate random bank clients**

At initialization, each client will generate public key (PK) and secret key (SK), and then call bank to register a client account for them. Then for registration, the bank will collect PK and clientId from client, and randomly assign deposit and withdrawal limits for that client.
 - **Step2. Generate digital signature**

The simulator will randomly ask client to randomly generate a message and a digital signature, until it has reach the requested number of (message, digital signature) pairs. The simulator will also randomly generate a fraction of message without digital signature.
 - **Step3. Verify transactions**

Using public keys to decode the digital signature collected in step 2, and compare the result with original message to verify the transaction. If the transaction is verified, the simulator will ask bank to further validate whether the requested transaction amount is within the deposit and/or withdrawal limits
- 4) One main simulation class: **InputParser.java** that checks the validity of command line arguments, and throw appropriate exception.