CCPG

Credit Card Payment Gateway

Prepared by

Shashank Venkatesha (RA2110026010269)
Tharun Raj (RA2110026010243)

GitHub Link:-
https://github.com/ShashankVenkatesha/OODP_UML_Project_CCPG

2022

Description

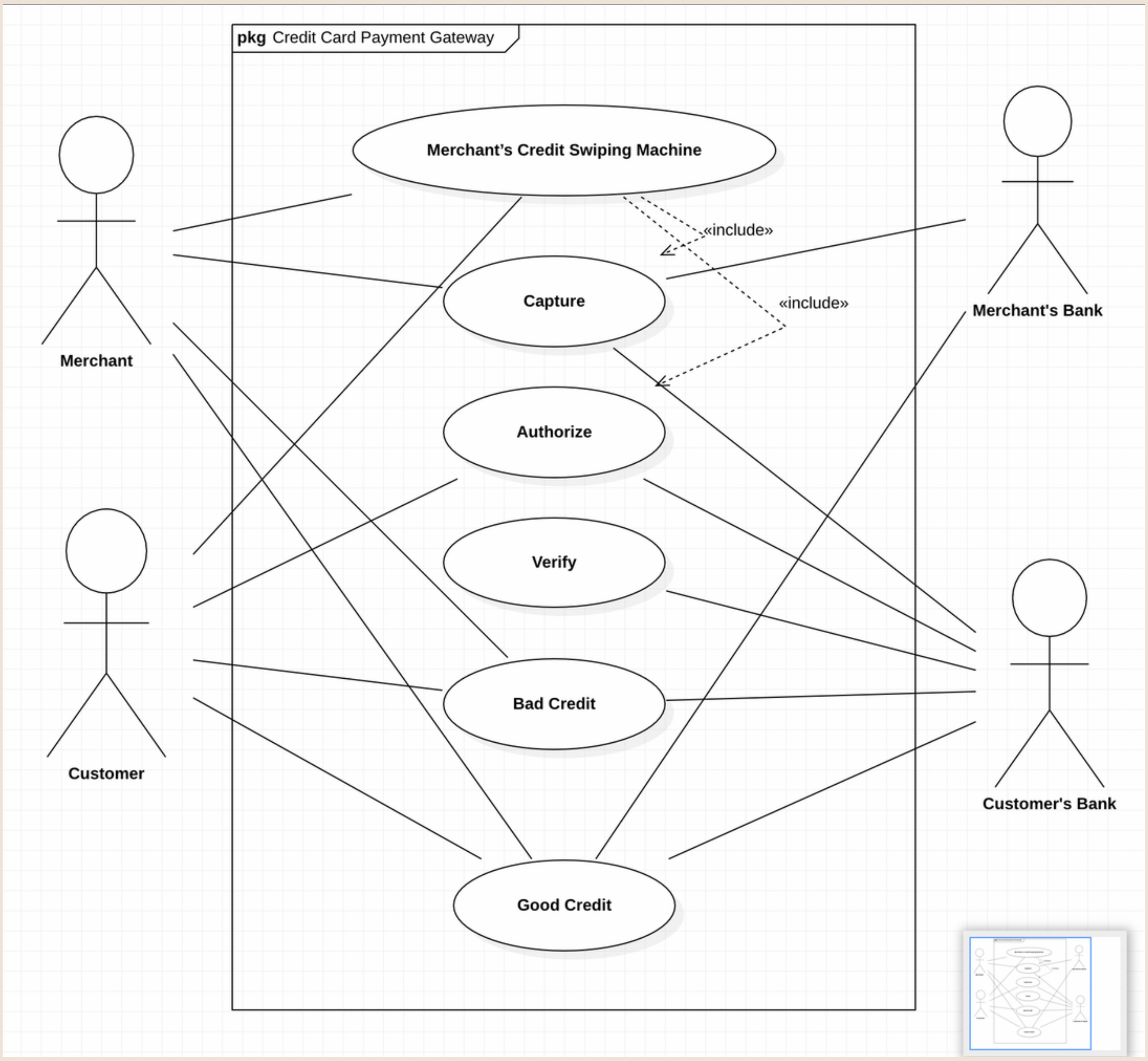
Payment gateway: A payment gateway is a software that transmits transaction information from a payment portal such as an e-commerce website to bank's processor and authorisation responses from the credit card issuing banks to the payment portal. It usually facilitates communication between the banks.

The payment gateway process flow

1. At checkout, merchant’s Card Swiping Machine sends encrypted payment details to payment gateway.
2. Payment gateway sends request to customer’s bank for authorisation
3. If payment was authorised, the customer’s bank pays the merchant

CCPG

Use Case Diagram

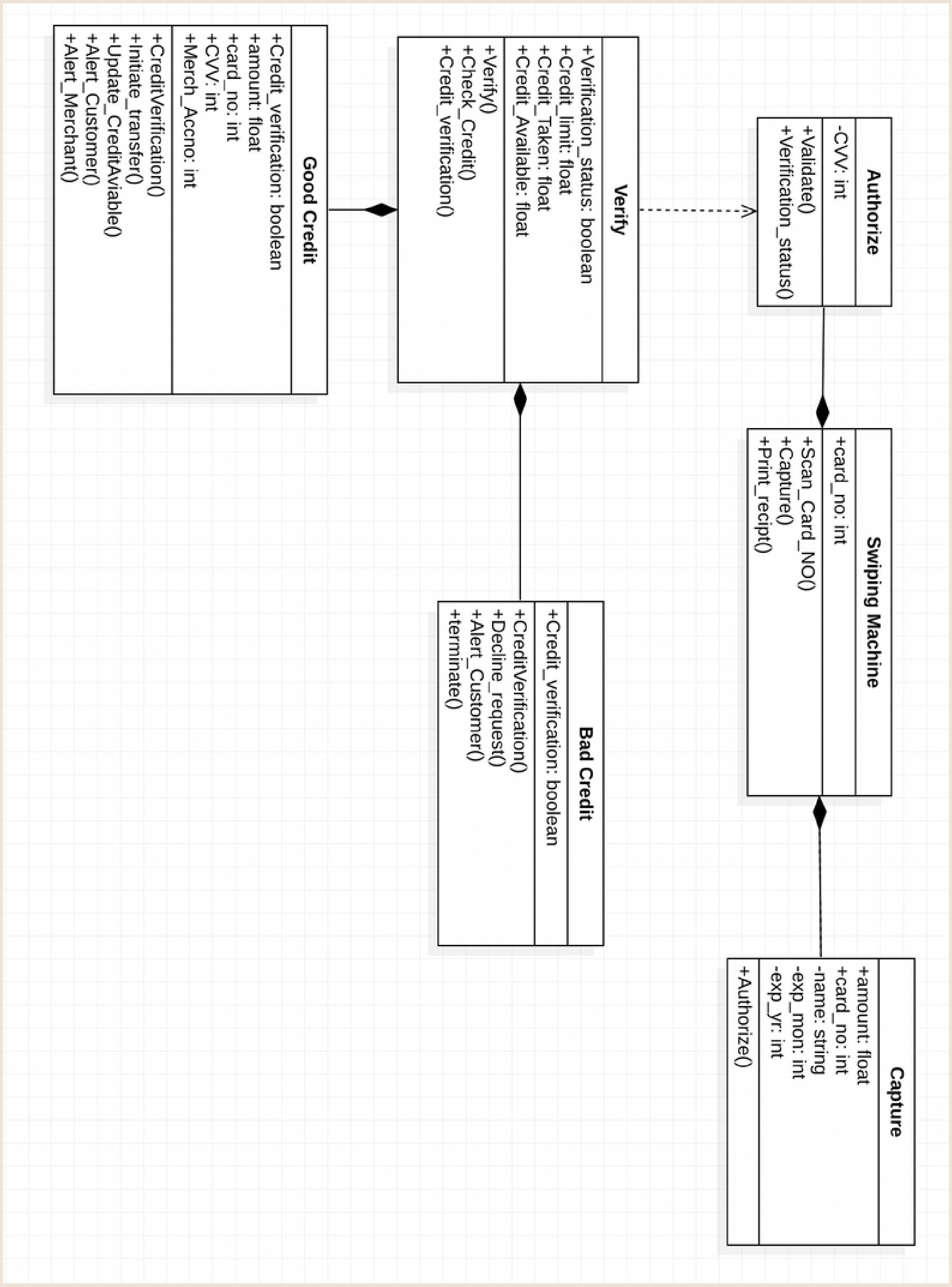


https://github.com/ShashankVenkatesha/OOD
P_UML_Project_CCPG

2022

CCPG

Class Diagram



https://github.com/ShashankVenkatesha/OOD
P_UML_Project_CCPG

2022

Relationships

Merchant's Card Swiping Machine use case is the most common type of credit card transaction. The requested amount of money should be first authorized by Customer's Credit Card Bank, and if approved, is further submitted for settlement. During the settlement funds approved for the credit card transaction are deposited into the Merchant's Bank account.

Capture use case describes several scenarios when merchant needs to complete some previously authorized transaction - either submitted through the payment gateway or requested without using the system, e.g. using voice authorization.

Verify use case describes zero or small amount verification transactions which could also include verification of some client's data such as address.

Good Credit use case describes situations when customer should receive a refund for a transaction that was either successfully processed and settled through the system or for some transaction that was not originally submitted through the payment gateway.

Bad Credit use case describes cases when it is needed to cancel one or several related transactions that were not yet settled. If possible, the transactions will not be sent for settlement. If the Void transaction fails, the original transaction is likely already settled.

Code

```
#ifndef _SWIPING MACHINE_H
#define _SWIPING MACHINE_H

class Swiping Machine {
public:
   int card_no;

void Scan_Card_NO();

void Capture();

void Print_recipt();
};

#endif //_SWIPING MACHINE_H
#ifndef _CAPTURE_H
#define _CAPTURE_H

class Capture {
public:
   float amount;
   int card_no;

void Authorize();
private:
   string name;
   int exp_mon;
   int exp_yr;
};
```

Code

```
#endif //_CAPTURE_H
#ifndef _AUTHORIZE_H
#define _AUTHORIZE_H

class Authorize {
public:

void Validate();

void Verification_status();
private:
   int CVV;
};

#endif //_AUTHORIZE_H
#ifndef _VERIFY_H
#define _VERIFY_H

class Verify {
public:
   boolean Verification_status;
   float Credit_limit;
   float Credit_Taken;
   float Credit_Available;

void Verify();

void Check_Credit();
```

Code

```
void Credit_verification();
};

#endif //_VERIFY_H
#ifndef _GOOD CREDIT_H
#define _GOOD CREDIT_H

class Good Credit {
public:
   boolean Credit_verification;
   float amount;
   int card_no;
   int CVV;
   int Merch_Accno;

void CreditVerification();

void Initiate_transfer();

void Update_CreditAviable();

void Alert_Customer();

void Alert_Merchant();
};

#endif //_GOOD CREDIT_H
```

Code

```
#ifndef _BAD CREDIT_H
#define _BAD CREDIT_H

class Bad Credit {
public:
   boolean Credit_verification;

void CreditVerification();

void Decline_request();

void Alert_Customer();

void terminate();
};

#endif //_BAD CREDIT_H
```

CCPG

Thank You

https://github.com/ShashankVenkatesha/OOD
P_UML_Project_CCPG

2022