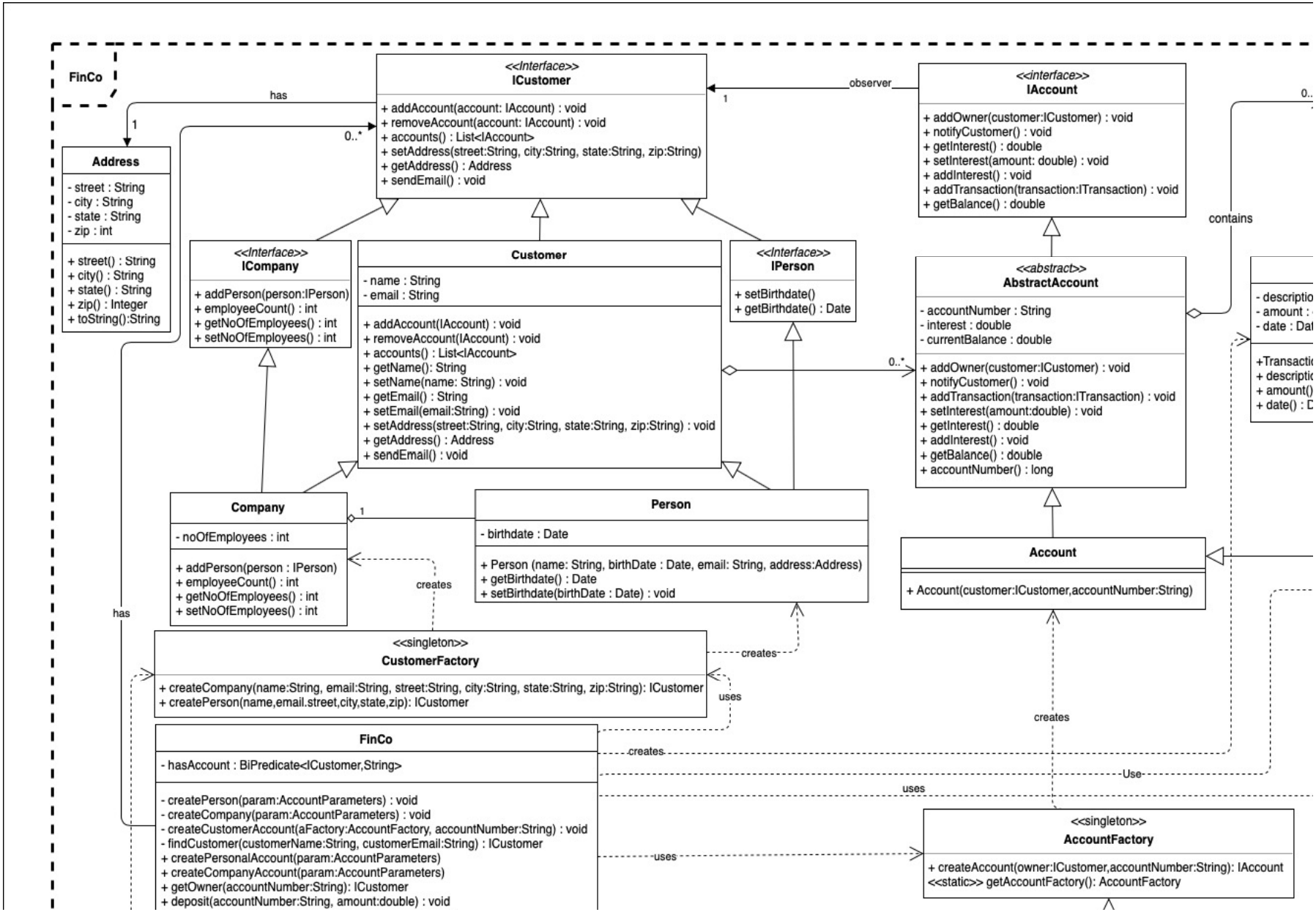
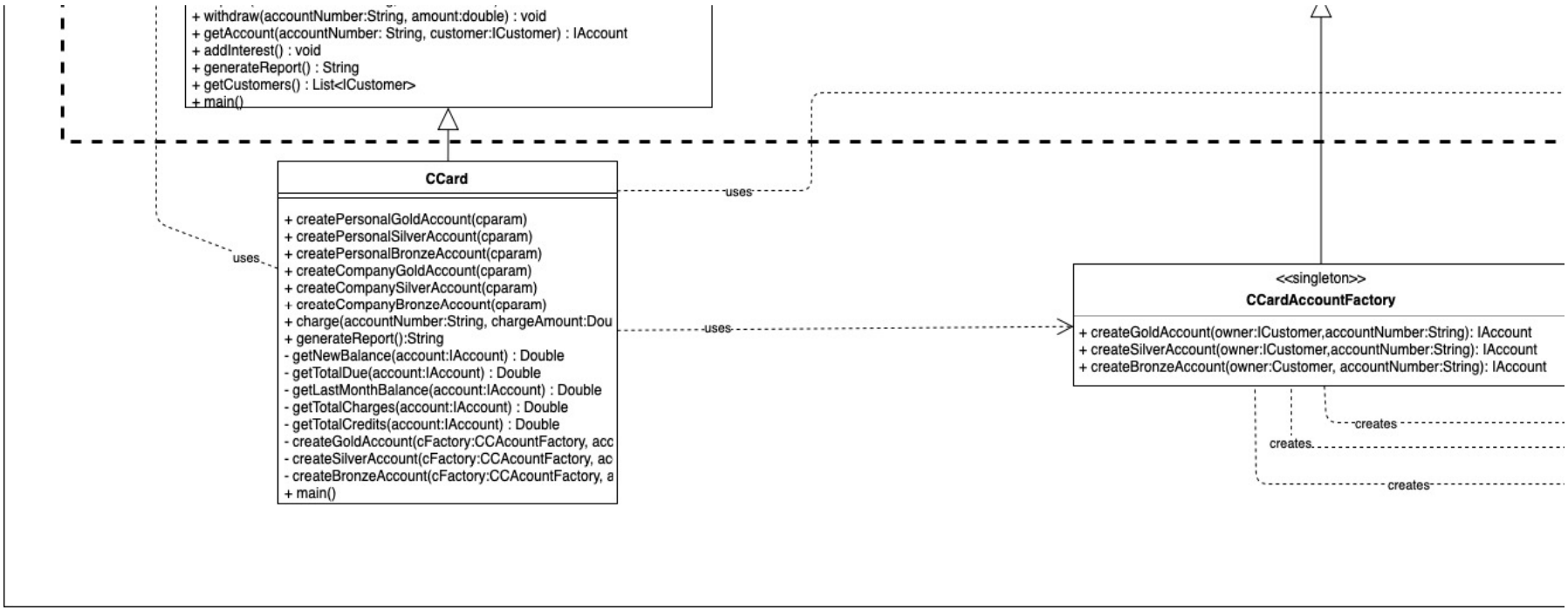
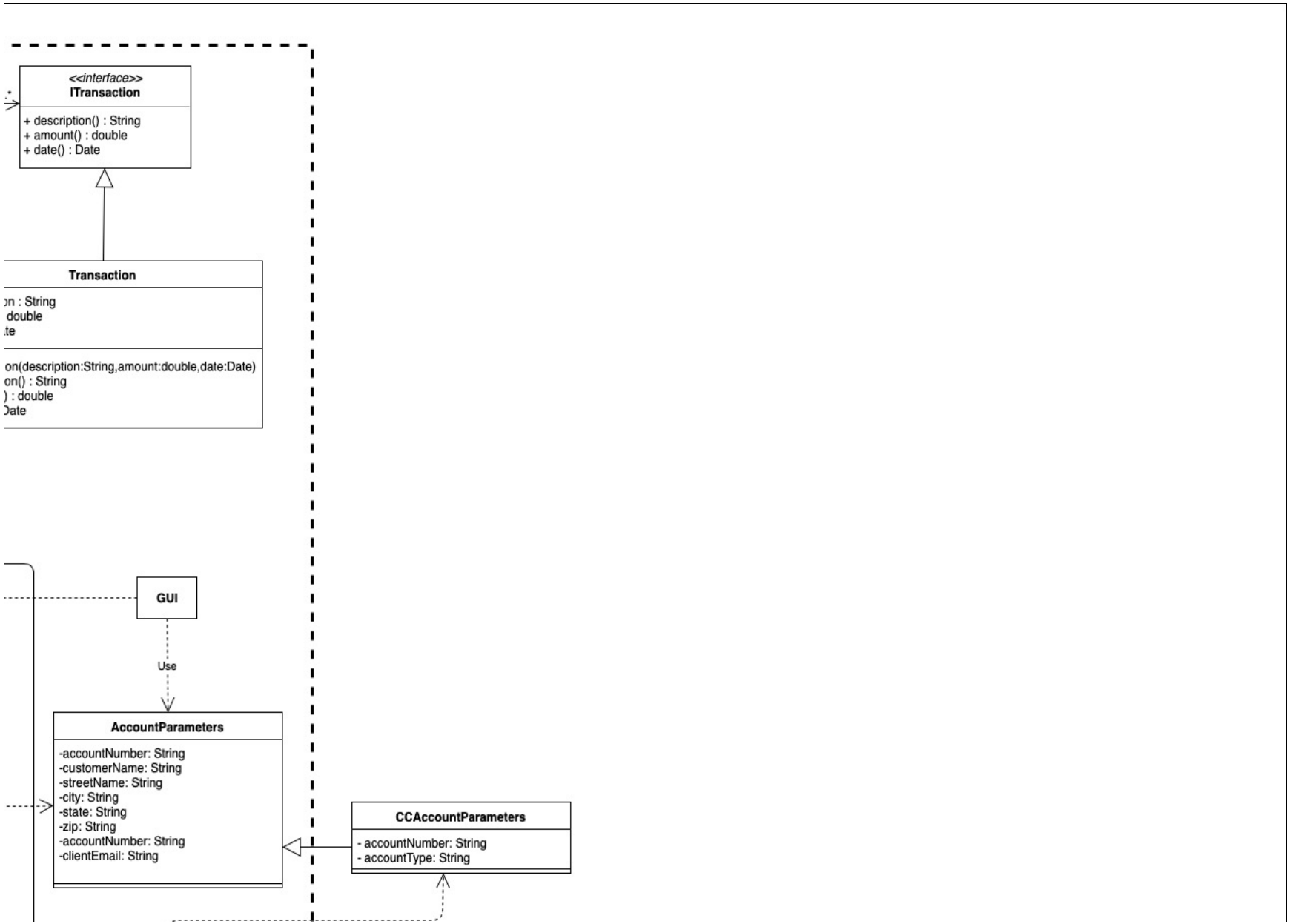
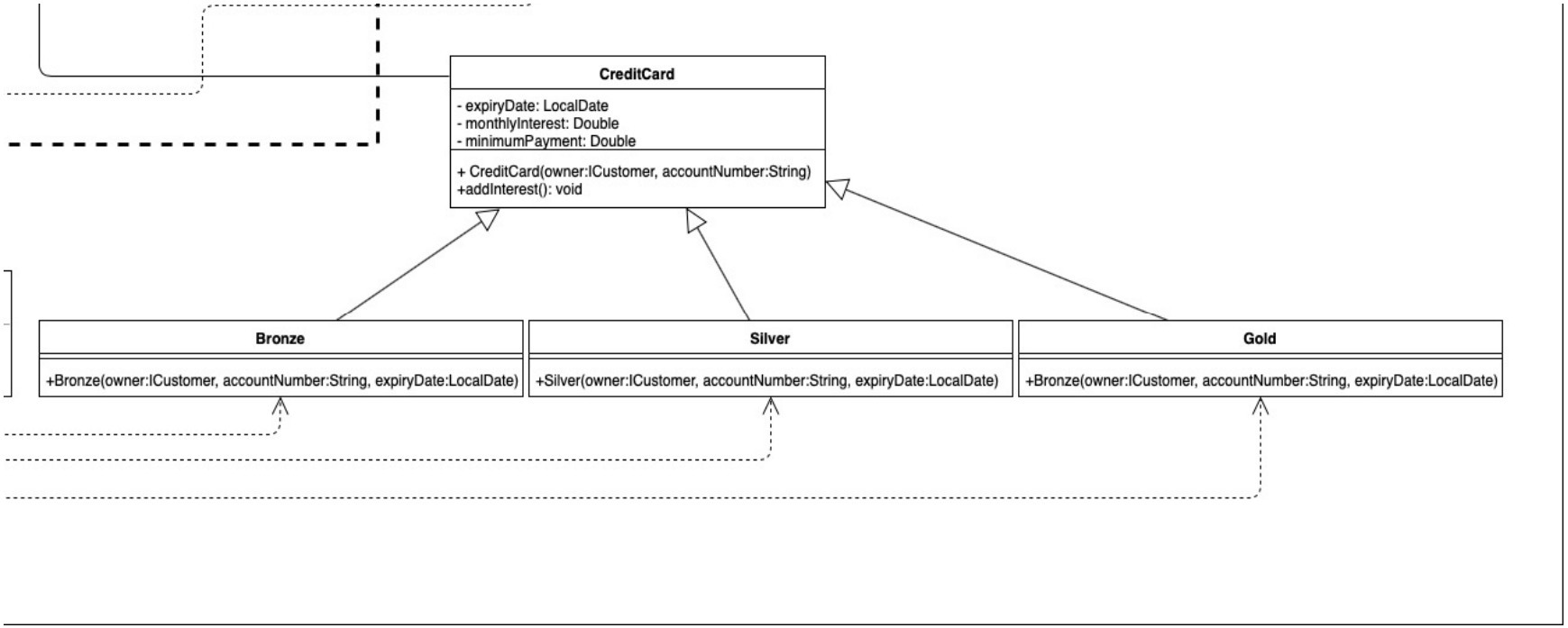
CCard - Class Diagram

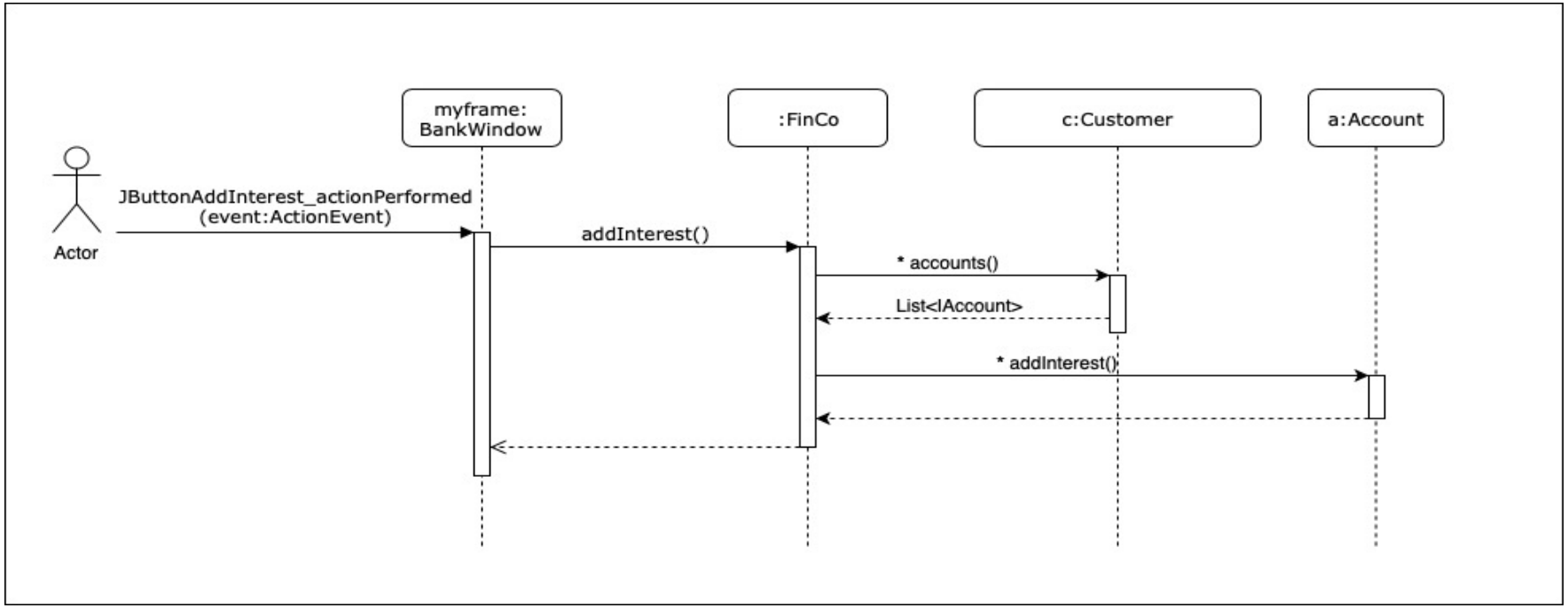



+ withdraw(accountNumber:String, amount:double) : void
+ getAccount(accountNumber: String, customer:ICustomer) : IAccount
+ addInterest() : void
+ generateReport() : String
+ getCustomers() : List<ICustomer>
+ main()
CCard
+ createPersonalGoldAccount(cparam)
+ createPersonalSilverAccount(cparam)
+ createPersonalBronzeAccount(cparam)
+ createCompanyGoldAccount(cparam)
+ createCompanySilverAccount(cparam)
+ createCompanyBronzeAccount(cparam)
+ charge(accountNumber:String, chargeAmount:Dou
+ generateReport():String
- getNewBalance(account:IAccount) : Double
- getTotalDue(account:IAccount) : Double
- getLastMonthBalance(account:IAccount) : Double
- getTotalCharges(account:IAccount) : Double
- getTotalCredits(account:IAccount) : Double
- createGoldAccount(cFactory:CCAcountFactory, acc
- createSilverAccount(cFactory:CCAcountFactory, ac
- createBronzeAccount(cFactory:CCAcountFactory, a
+ main()
uses
uses
uses
<<singleton>>
CCardAccountFactory
+ createGoldAccount(owner:ICustomer,accountNumber:String): IAccount
+ createSilverAccount(owner:ICustomer,accountNumber:String): IAccount
+ createBronzeAccount(owner:Customer, accountNumber:String): IAccount
creates
creates
creates

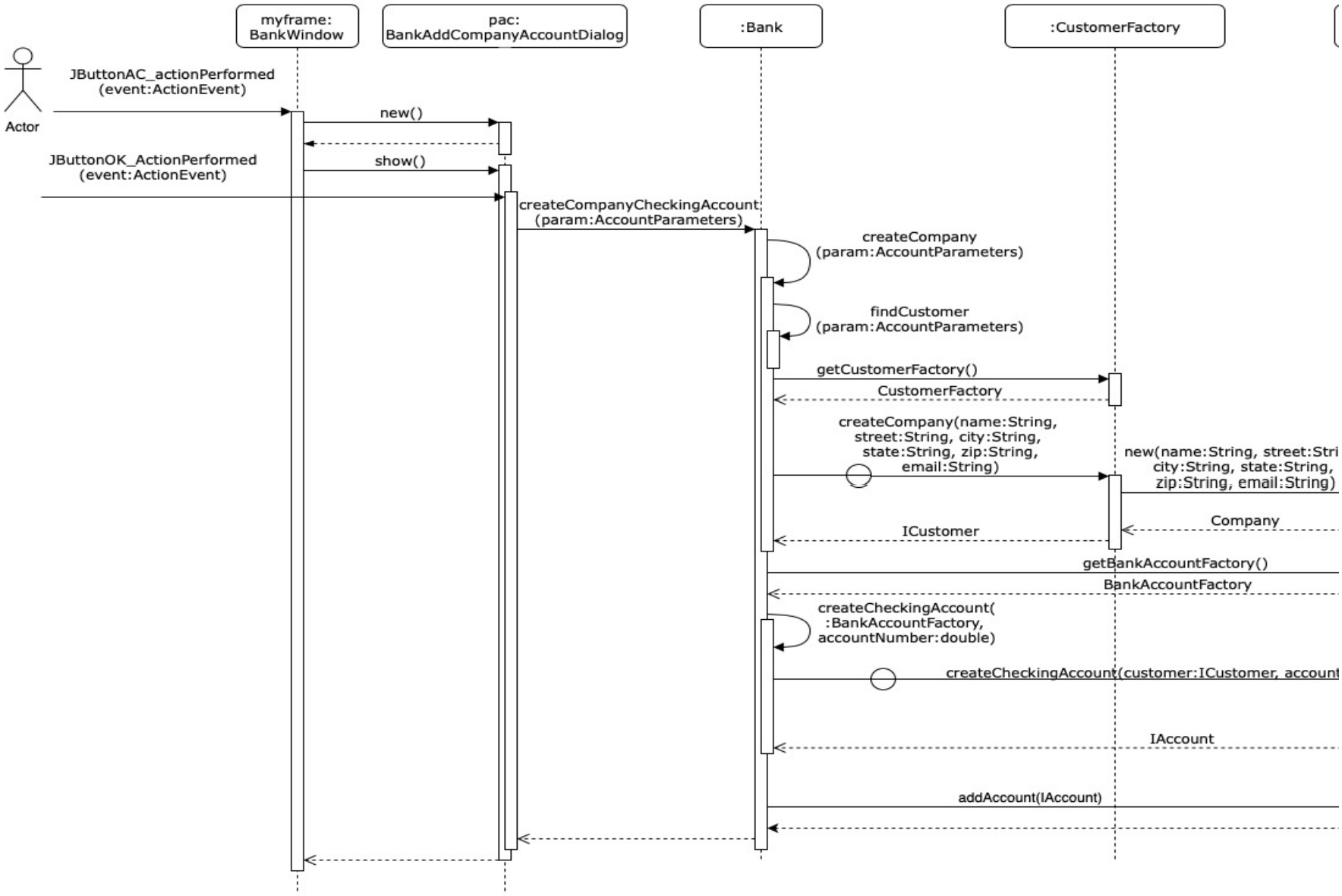

<<interface>>
ITransaction
+ description() : String
+ amount() : double
+ date() : Date
Transaction
on(description:String,amount:double,date:Date)
GUI
Use
AccountParameters
-accountNumber: String
-customerName: String
-streetName: String
-city: String
-state: String
-zip: String
-accountNumber: String
-clientEmail: String
CCAccountParameters
- accountNumber: String
- accountType: String

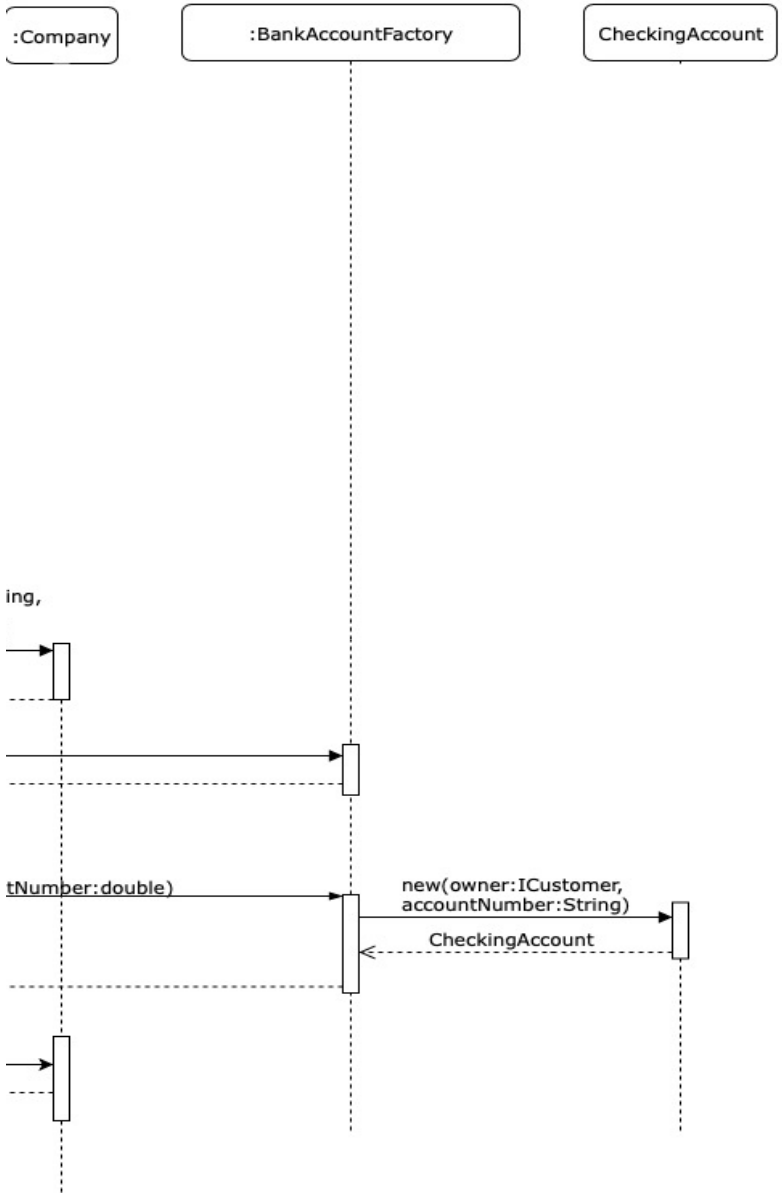

CreditCard
- expiryDate: LocalDate
- monthlyInterest: Double
- minimumPayment: Double
+ CreditCard(owner:ICustomer, accountNumber:String)
+addInterest(): void
Bronze
+Bronze(owner:ICustomer, accountNumber:String, expiryDate:LocalDate)
Silver
+Silver(owner:ICustomer, accountNumber:String, expiryDate:LocalDate)
Gold
+Bronze(owner:ICustomer, accountNumber:String, expiryDate:LocalDate)

Bank - Add Interest - Sequence Diagram

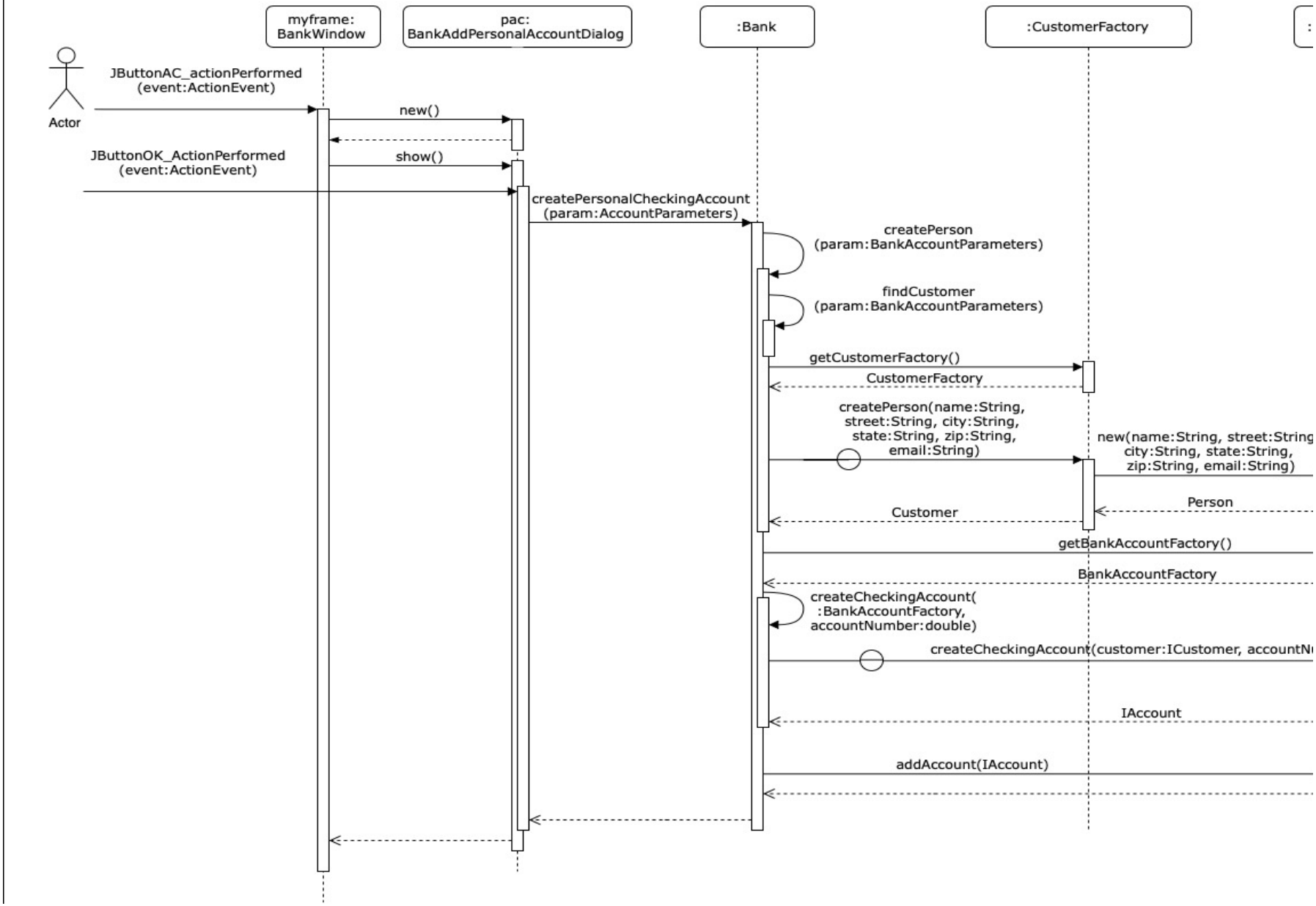


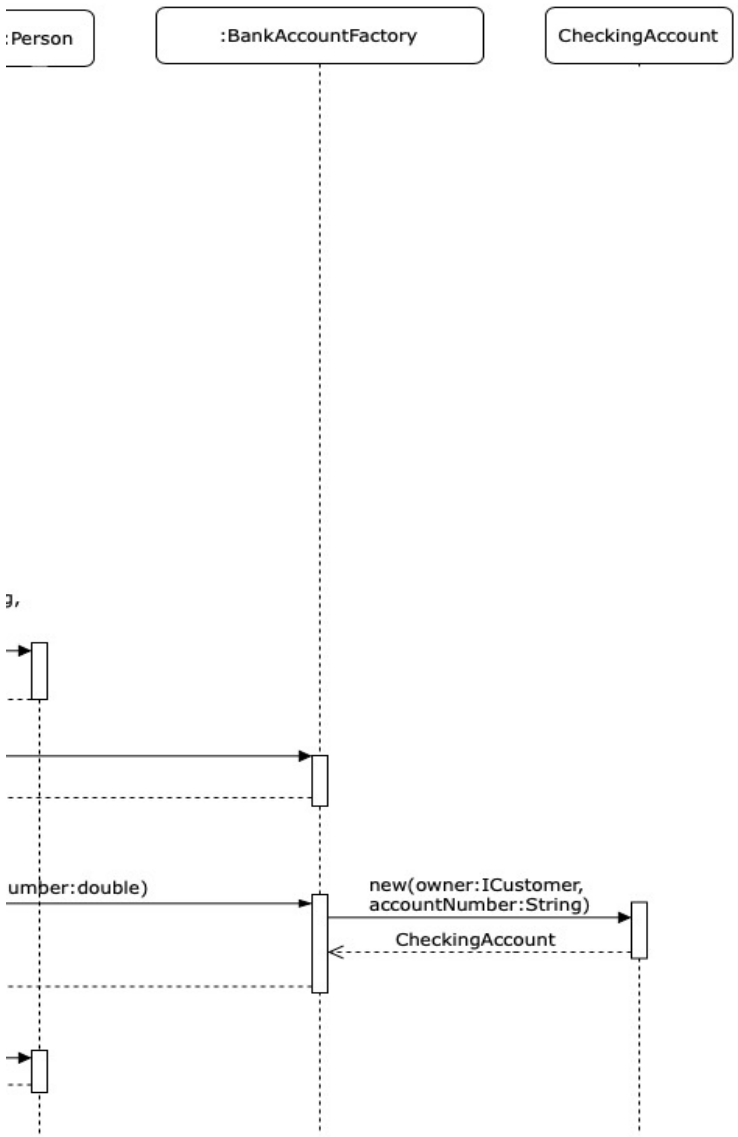
Bank - Create Company Checking Account - Sequence Diagram



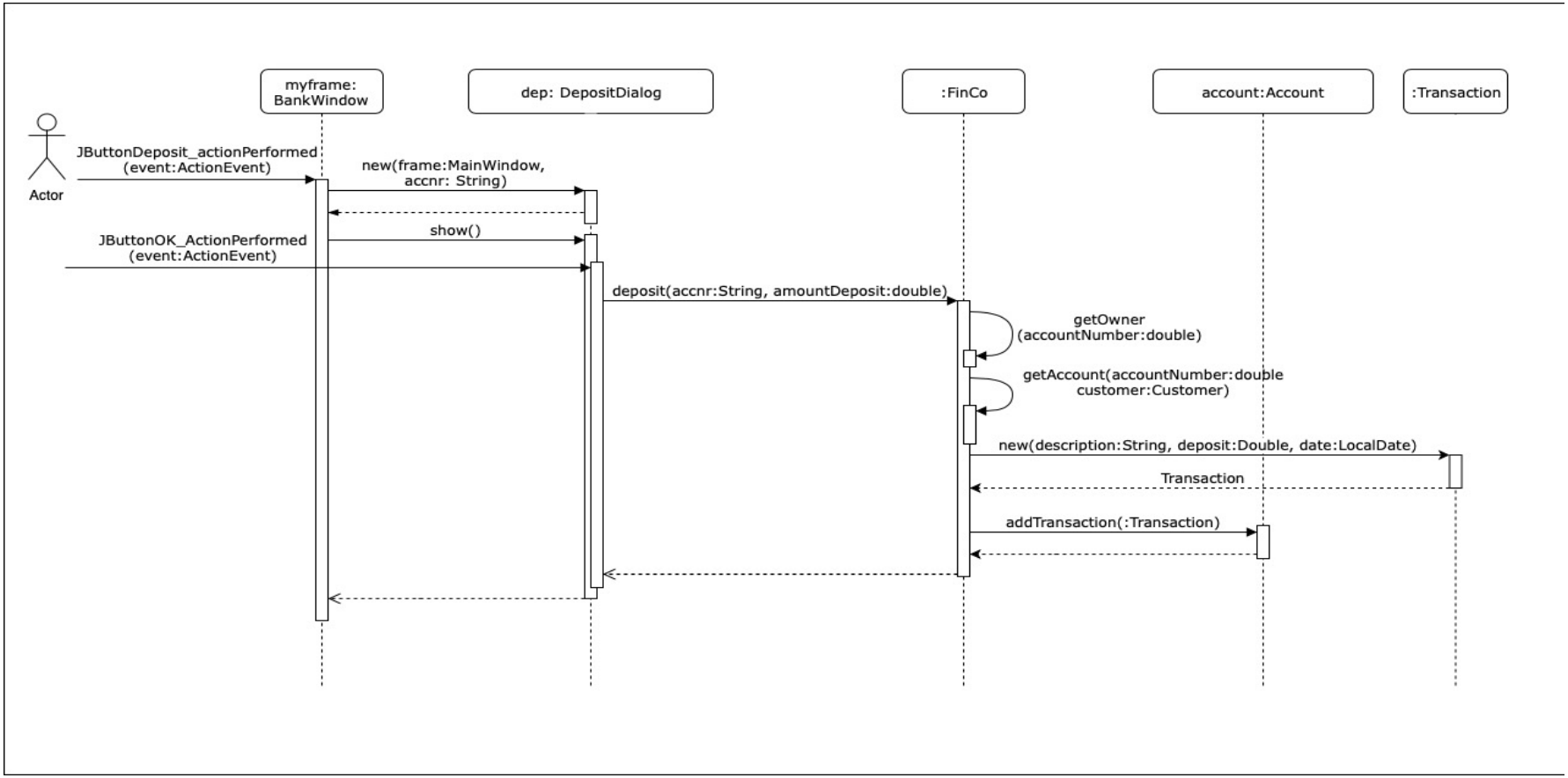

:Company
:BankAccountFactory
CheckingAccount
ing,
tNumber:double)
new(owner:ICustomer,
accountNumber:String)
CheckingAccount

Bank - Create Personal Checking Account - Sequence Diagram

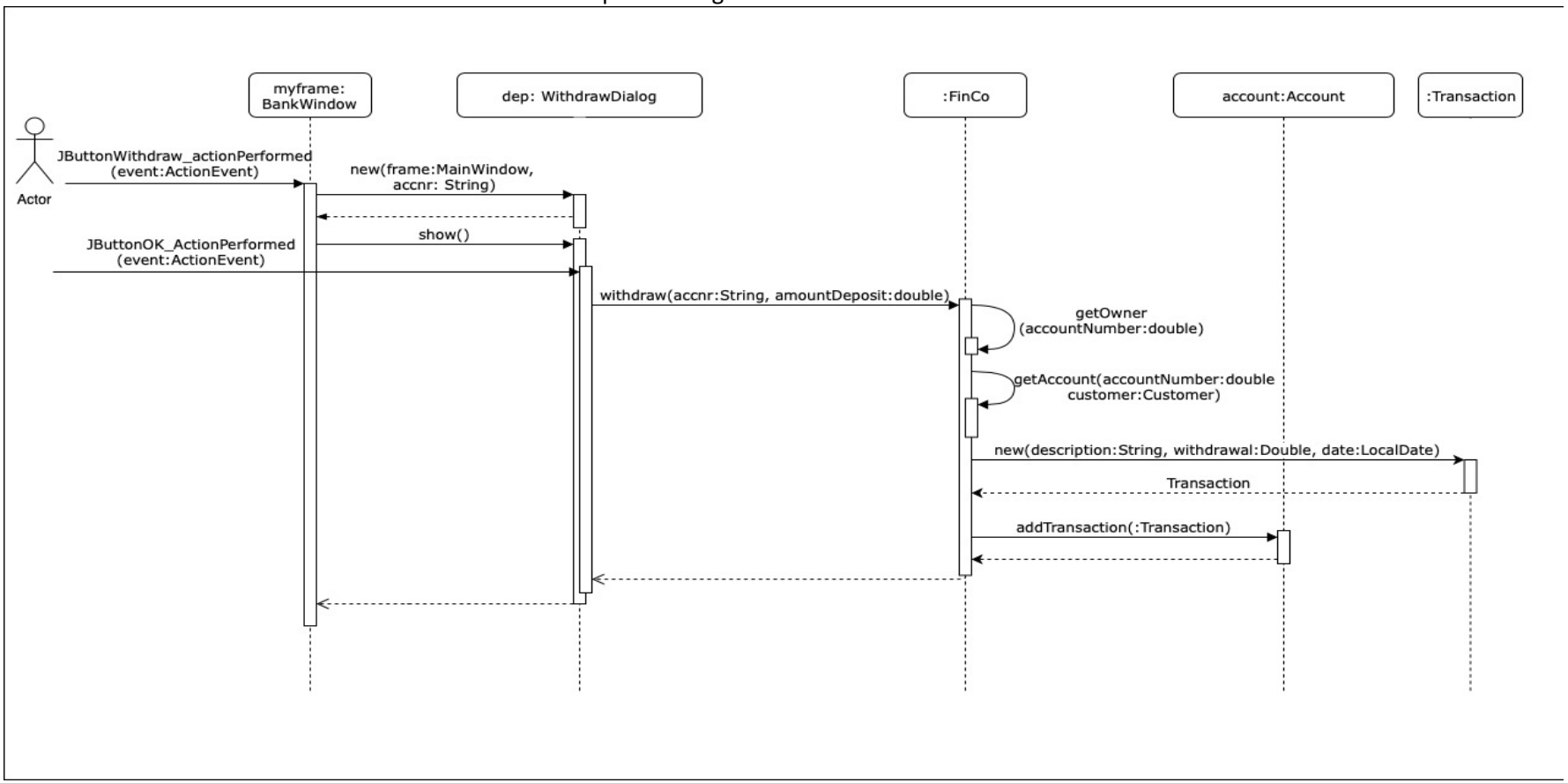



:Person
:BankAccountFactory
CheckingAccount
umber:double)
new(owner:ICustomer,
accountNumber:String)
CheckingAccount

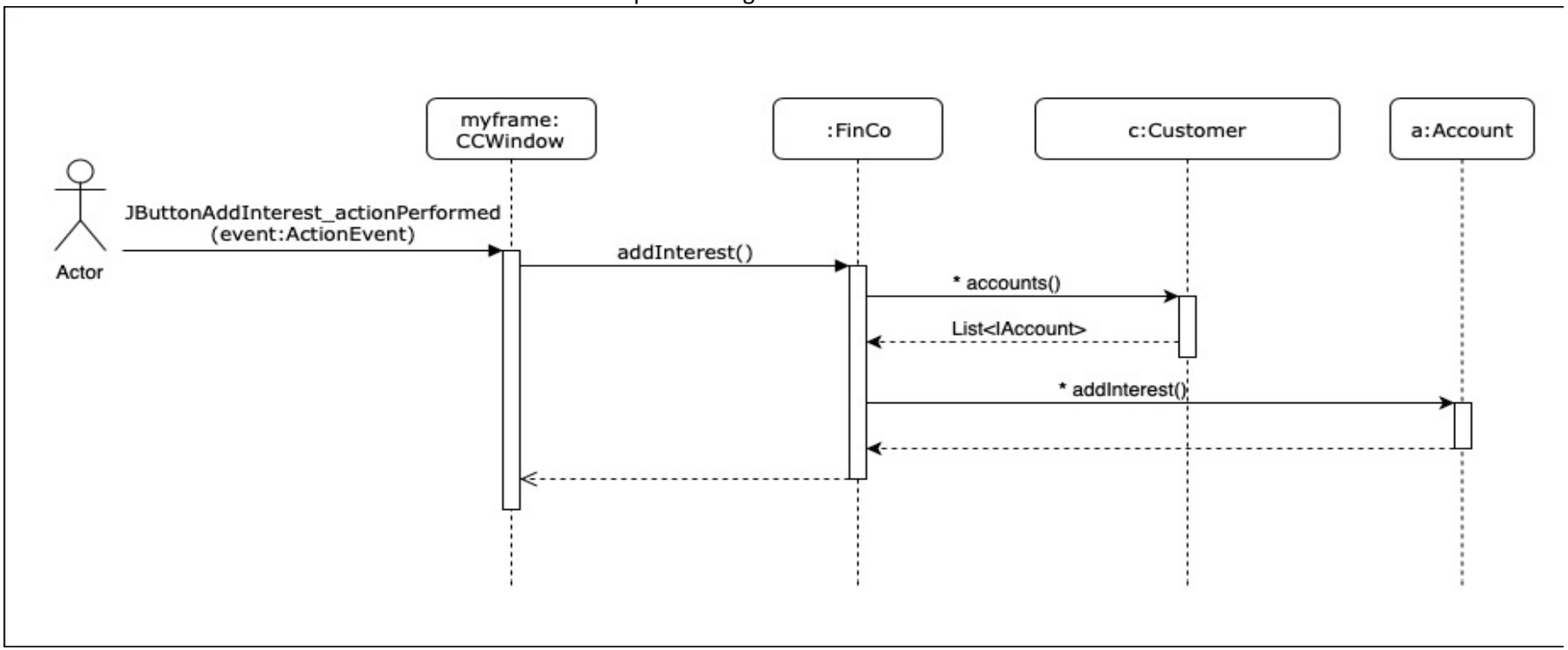
Bank - Deposit - Sequence Diagram



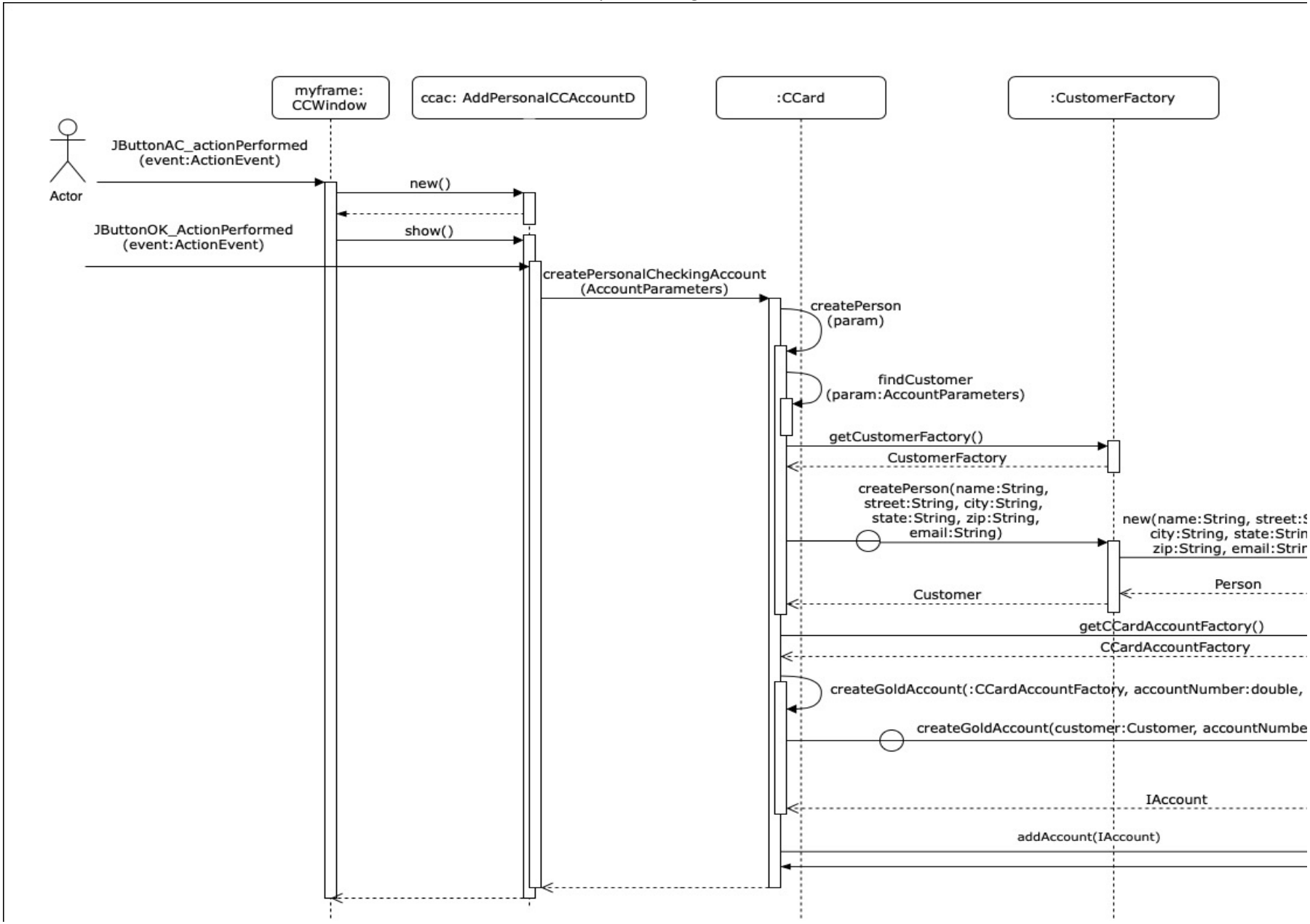
Bank - Withdraw - Sequence Diagram

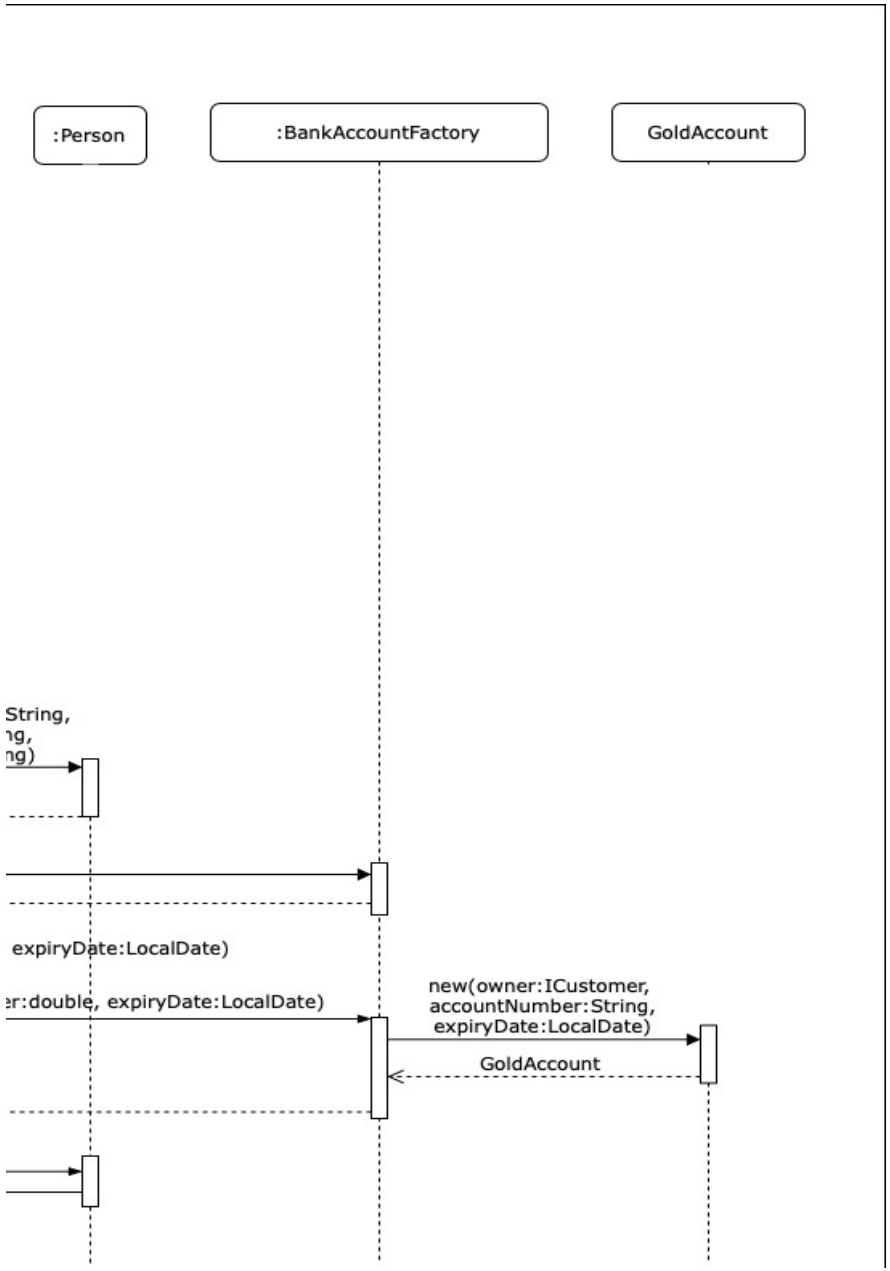


CCard - Add Interest - Sequence Diagram

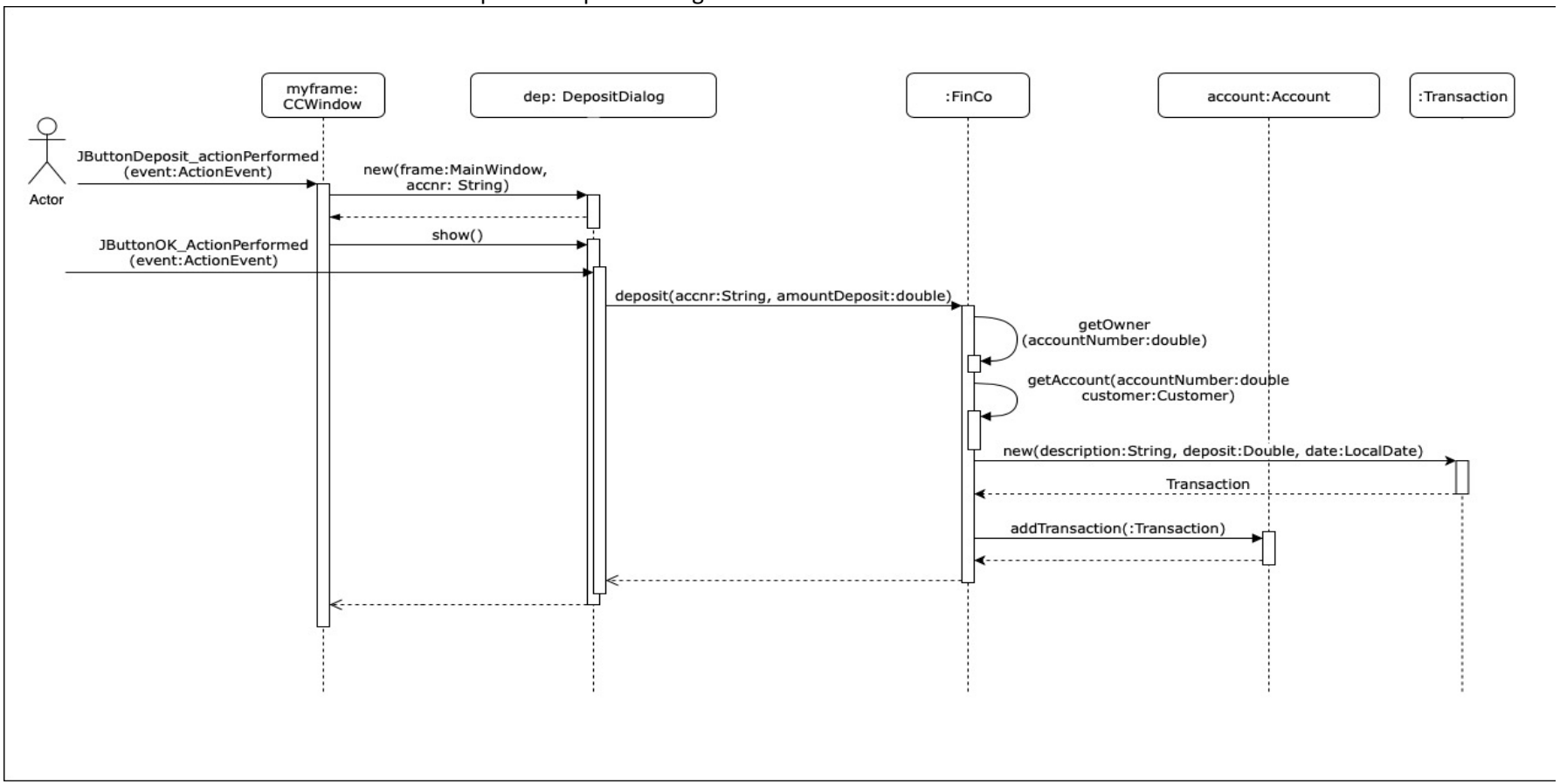


CCard - Create Gold Account - Sequence Diagram

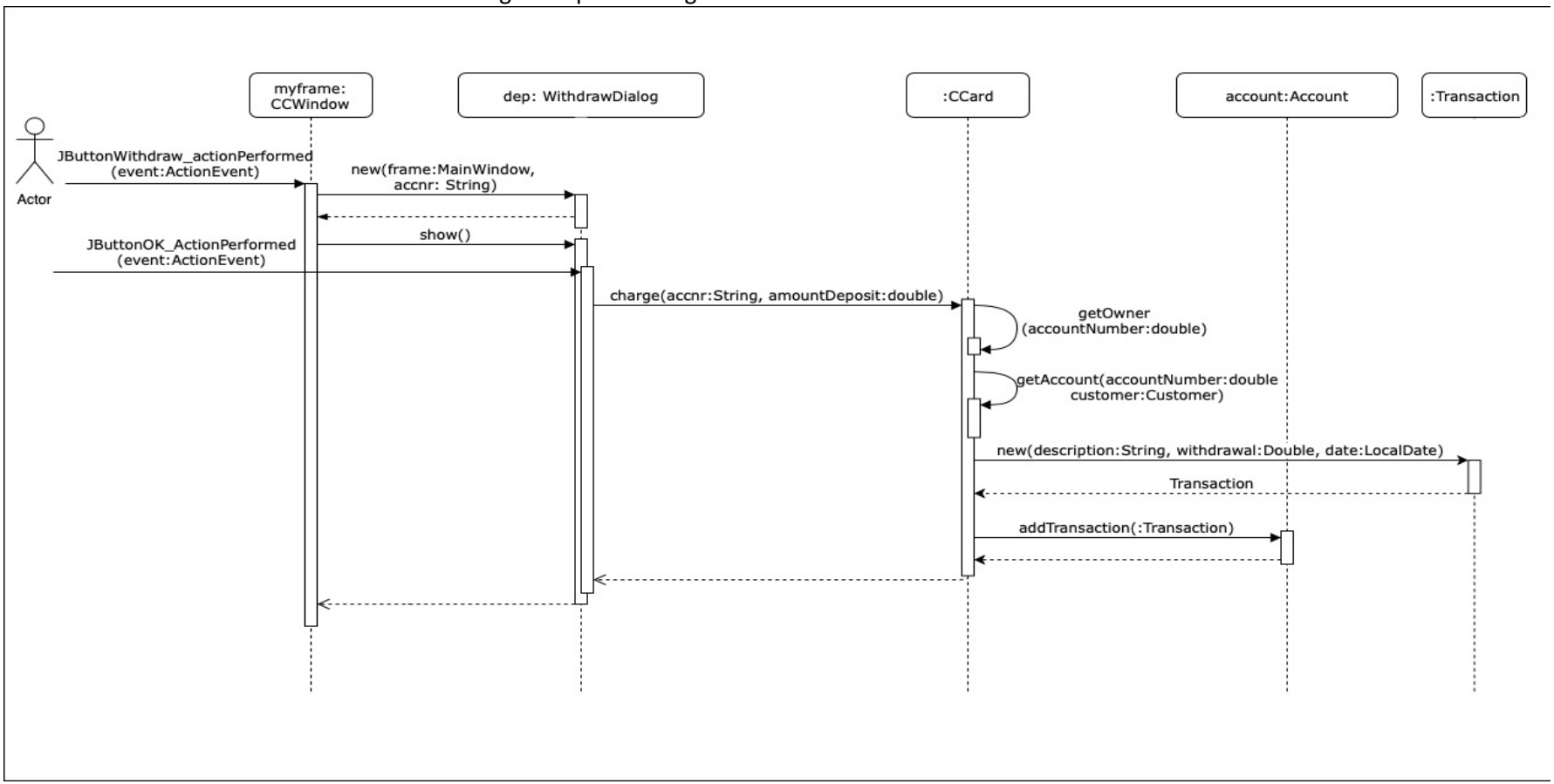



:Person
:BankAccountFactory
GoldAccount
String,
ng,
ng)
expiryDate:LocalDate)
er:double, expiryDate:LocalDate)
new(owner:ICustomer,
accountNumber:String,
expiryDate:LocalDate)
GoldAccount

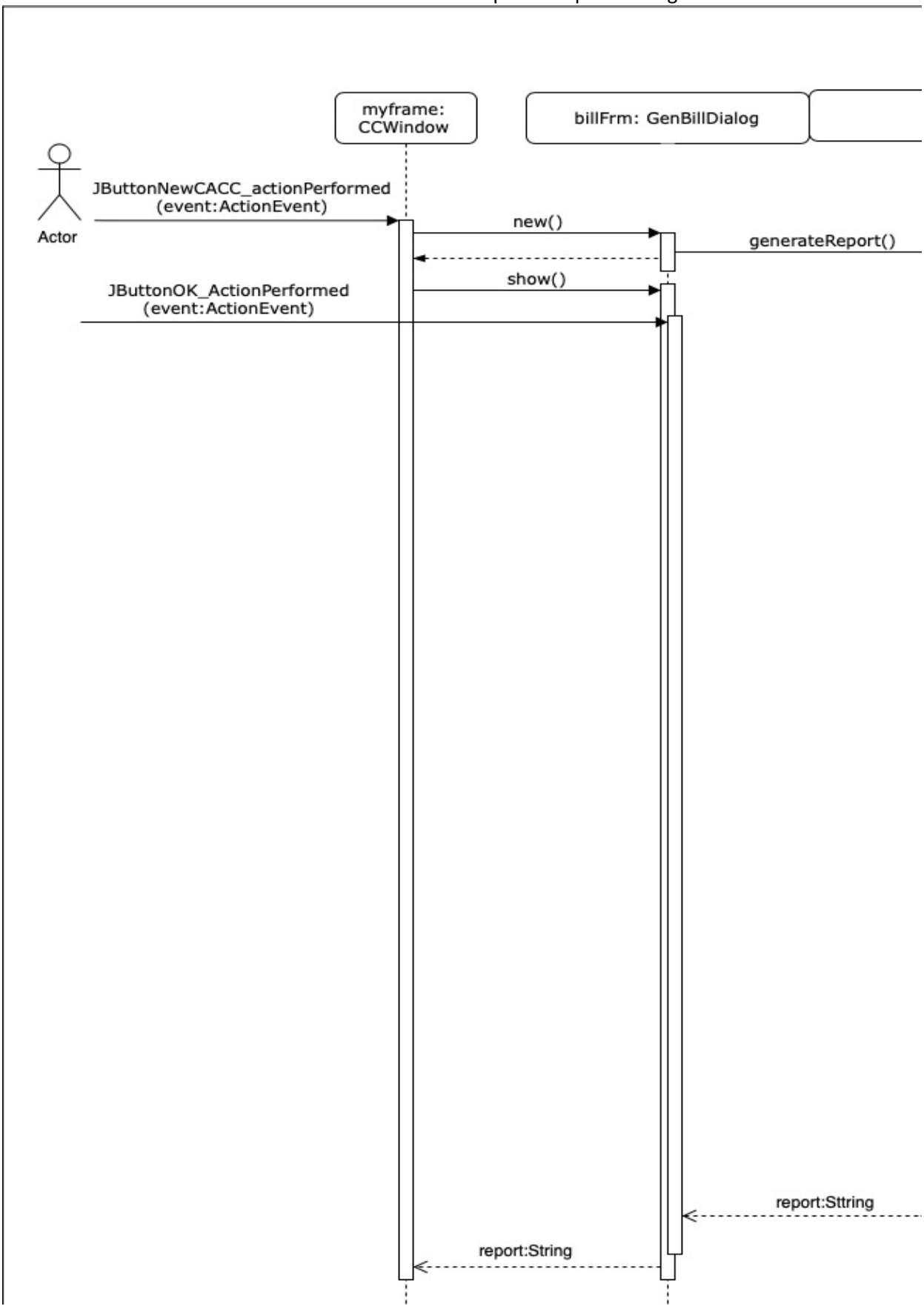
CCard - Deposit - Sequence Diagram

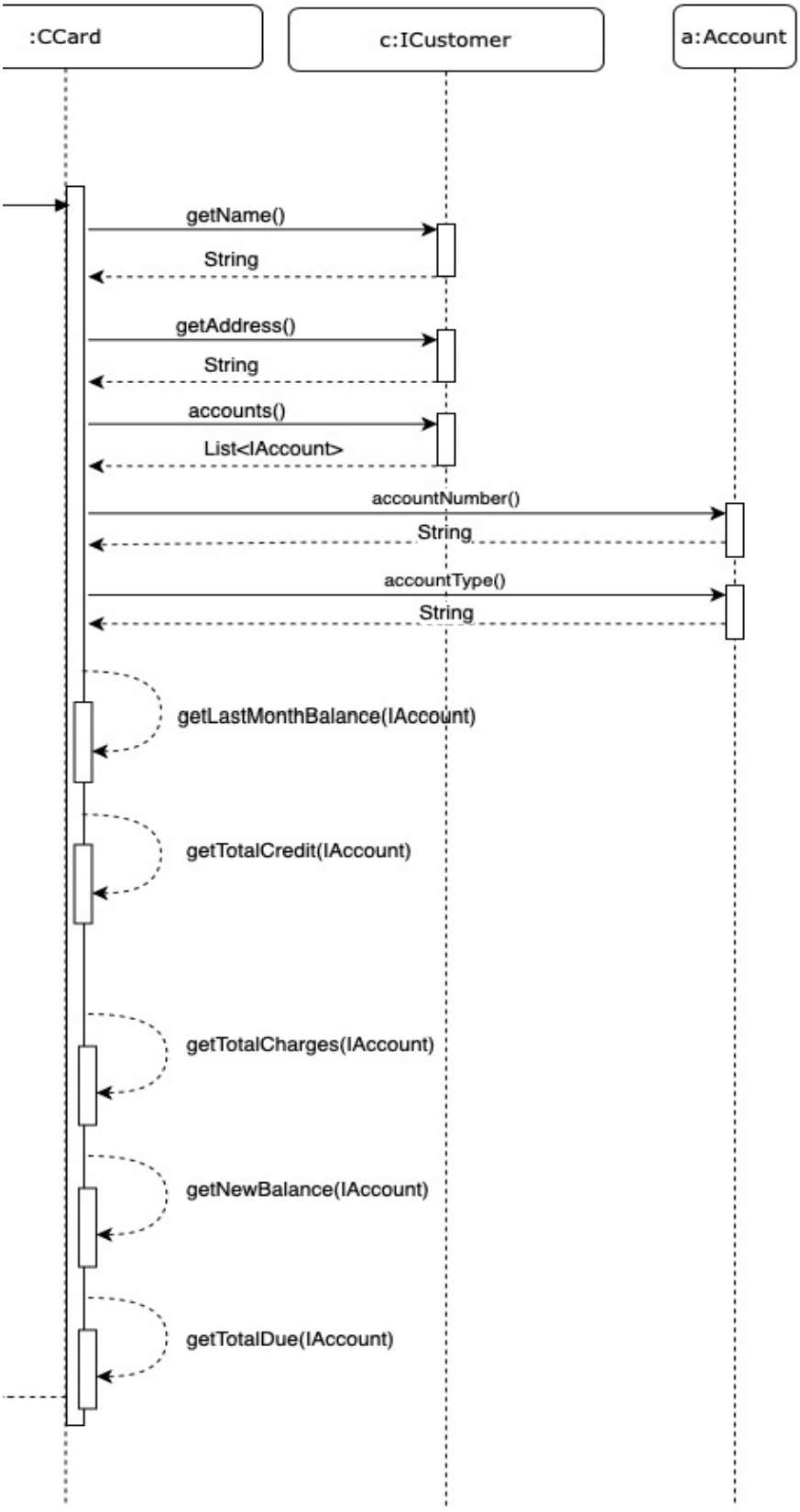


CCard - Charge - Sequence Diagram



CCard - Generate Report - Sequence Diagram




:CCard
c:ICustomer
a:Account
getName()
String
getAddress()
String
accounts()
List<IAccount>
accountNumber()
String
accountType()
String
getLastMonthBalance(IAccount)
getTotalCredit(IAccount)
getTotalCharges(IAccount)
getNewBalance(IAccount)
getTotalDue(IAccount)