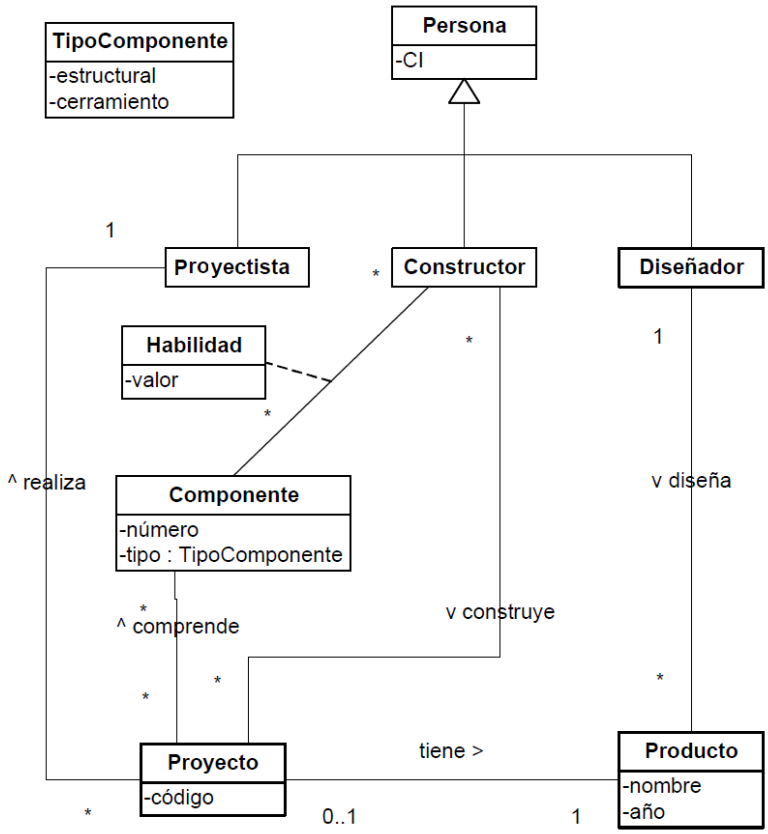
Programación Avanzada

SOLUCION PRIMER PARCIAL 2015

Problema 1

Modelo de Dominio

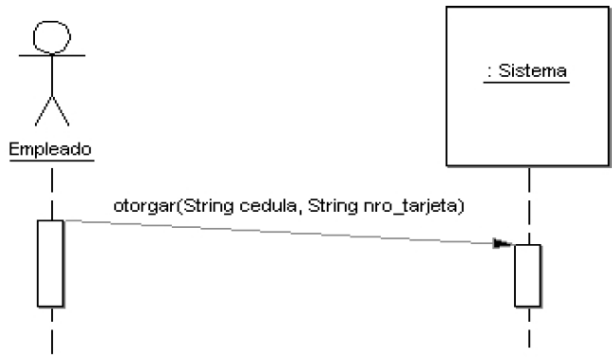


Restricciones

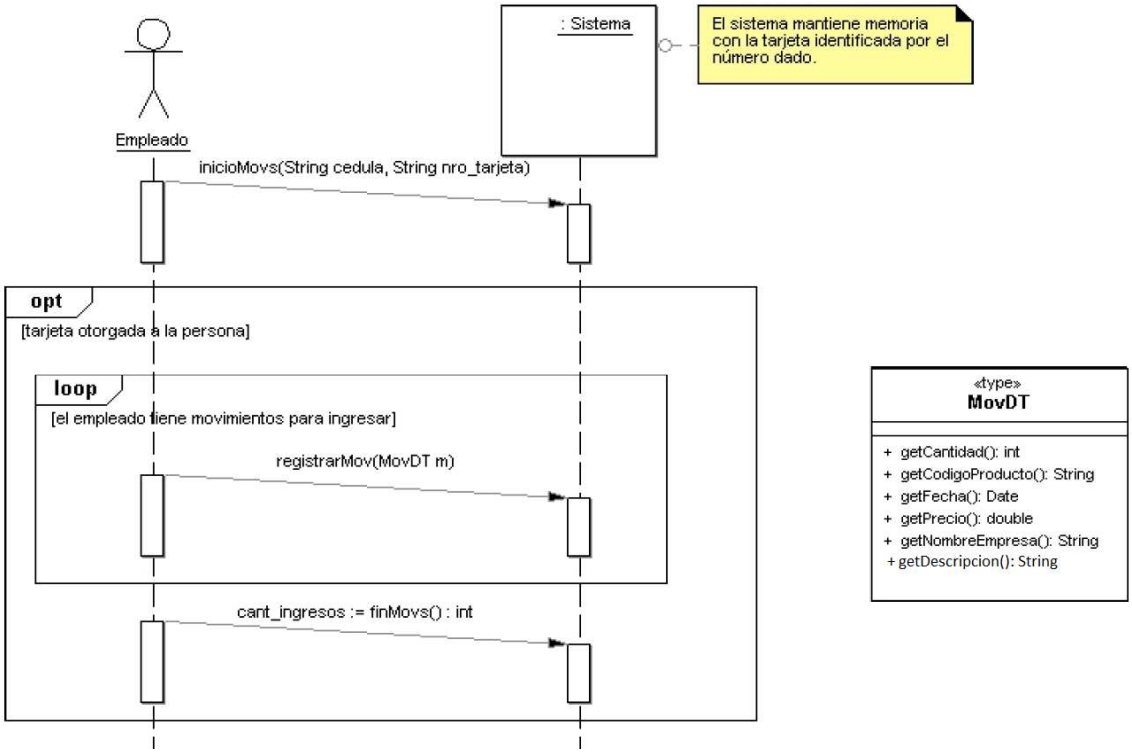
- Una Persona es identificada por su CI.
- Un Componente es identificado por su número.
- Un Proyecto es identificado por su código.
- Un Producto es identificado por su nombre.
- El valor de una Habilidad pertenece al rango 0..5.
- Un Constructor puede construir un Proyecto si tiene valor de Habilidad mayor o igual a 2 en todos los Componentes del mismo.

Problema 2

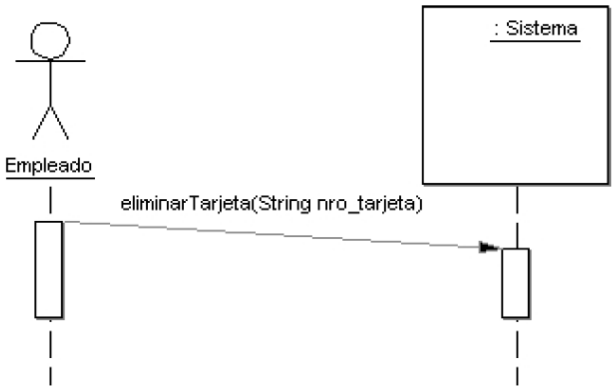
a) DSS Otorgamiento de tarjeta



DSS Ingreso de movimientos



DSS Baja de tarjeta



b) Contratos

void otorgar(cedula: String, nro_tarjeta: String)

pre: existe una instancia de Persona con cédula de identidad = cedula.
pre: existe una instancia de Tarjeta identificada por el número = nro_tarjeta.
pre: no existe un link de Otorgamiento entre la instancia de Persona con cédula de identidad = cedula y la instancia de Tarjeta identificada por el número = nro_tarjeta.
post: si existe un link de Solicitud entre la instancia de Persona con cédula de identidad = cedula y la instancia de Tarjeta identificada por el número = nro_tarjeta, entonces se elimina dicho link y se crea un link de Otorgamiento entre el mismo par de instancias.

bool inicioMovs(cedula: String, nro_tarjeta: String)

pre: existe una instancia de Persona con cédula de identidad = cedula.
pre: existe una instancia de Tarjeta identificada por el número = nro_tarjeta.
post: el sistema crea un link a la instancia de Tarjeta identificada por nro_tarjeta como la tarjeta actual.
post: el sistema mantiene en memoria la cantidad ingresada y le asigna el valor 0.
post: devuelve true si existe un link de Otorgamiento entre la instancia de Persona con cédula de identidad = cedula y la instancia de Tarjeta identificada por el número = nro_tarjeta. En caso contrario, devuelve false.

void registrarMov(m: MovDT)

pre: el sistema tiene una instancia de Tarjeta como tarjeta actual.
post: se crea una instancia de Movimiento con valor de atributo fecha = m.getFecha(), cant = m.getCantidad() y precioUnitMov = m.getPrecio().
post: se crea un link entre la tarjeta actual recordada por el sistema y la instancia de Movimiento creada.
post: se crea un link entre la instancia de Movimiento creada y la instancia de Producto identificada por el código = m.getCodigoProducto() y nombre de empresa = m.getNombreEmpresa().
post: si y sólo si no existe un instancia de Producto identificada por el código = m.getCodigoProducto() y nombre de empresa = m.getNombreEmpresa(), se crea una
instancia de Producto con valor de atributo codigo = m.getCodigoProducto(), nombreEmpresa = m.getNombreEmpresa(), descripcion = m.getDescripcion() y precioActual = m.getPrecio().
post: se incrementa en 1 la cantidad ingresada.

int finMovs()

pre: el sistema tiene una instancia de Tarjeta como tarjeta actual.
post: el sistema elimina el link a la instancia de Tarjeta identificada por nro_tarjeta como la tarjeta actual.
post: devuelve como resultado el valor de cantidad ingresada.

void eliminarTarjeta(nro_tarjeta: String)

pre: existe una instancia de Tarjeta identificada por el número = nro_tarjeta.
post: no existe instancia de Tarjeta identificada por el número = nro_tarjeta.
post: no existen instancias de Movimiento que estaban asociadas con la instancia de Tarjeta identificada por el número = nro_tarjeta.
post: se eliminan los links que existían entre las instancias de producto y las instancias de Movimiento que estaban asociadas con la instancia de Tarjeta identificada por el número = nro_tarjeta.
post: si existía link de Solicitud entre la instancia de Tarjeta identificada por el número = nro_tarjeta y una instancia de Persona, se lo elimina.
post: si existía link de Otorgamiento entre la instancia de Tarjeta identificada por el número = nro_tarjeta y una instancia de Persona, se lo elimina.

Problema 3

Fecha.h

```
class Fecha {

      private:
              int dia;
              int mes;
              int anio;

      public:
              Fecha();
              Fecha(int d, int m, int a);
              int getDia();
              int getMes();
              int getAnio();
              void setDia(int d);
              void setMes(int m);
              void setAnio(int a);
              ~Fecha();
};
```

Fecha.cpp

```
#include “Fecha.h”

Fecha::Fecha(){
    dia = 1;
    mes = 1;
    anio = 2015;

}

Fecha::Fecha(int d, int m, int a){
    dia = d;
    mes = m;
    anio = a;
}

int Fecha::getDia(){
    return dia;
}

int Fecha::getMes(){
    return mes;
}

int Fecha::getAnio(){
    return anio;
}

void Fecha::setDia(int d){
    dia = d;
}

void Fecha::setMes(int m){
    mes = m;
}

void Fecha::setAnio(int a){
    anio = a;
}

Fecha::~Fecha(){

}
```