

CÁTEDRA DE BASE DE DATOS AVANZADA

Universidad del Aconcagua

TRABAJO PRÁCTICO

N° 4

Sigal Nicolás – 110994

CAJERO AUTOMÁTICO

DIAGRAMA ENTIDAD-RELACION

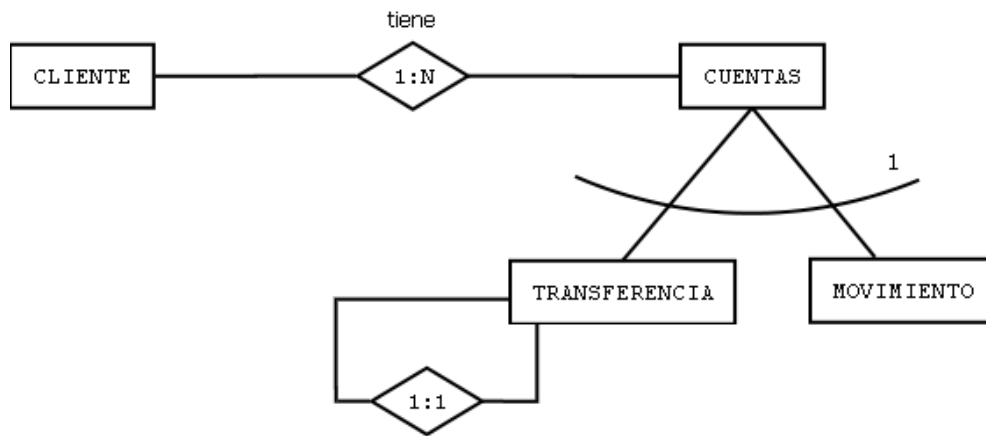
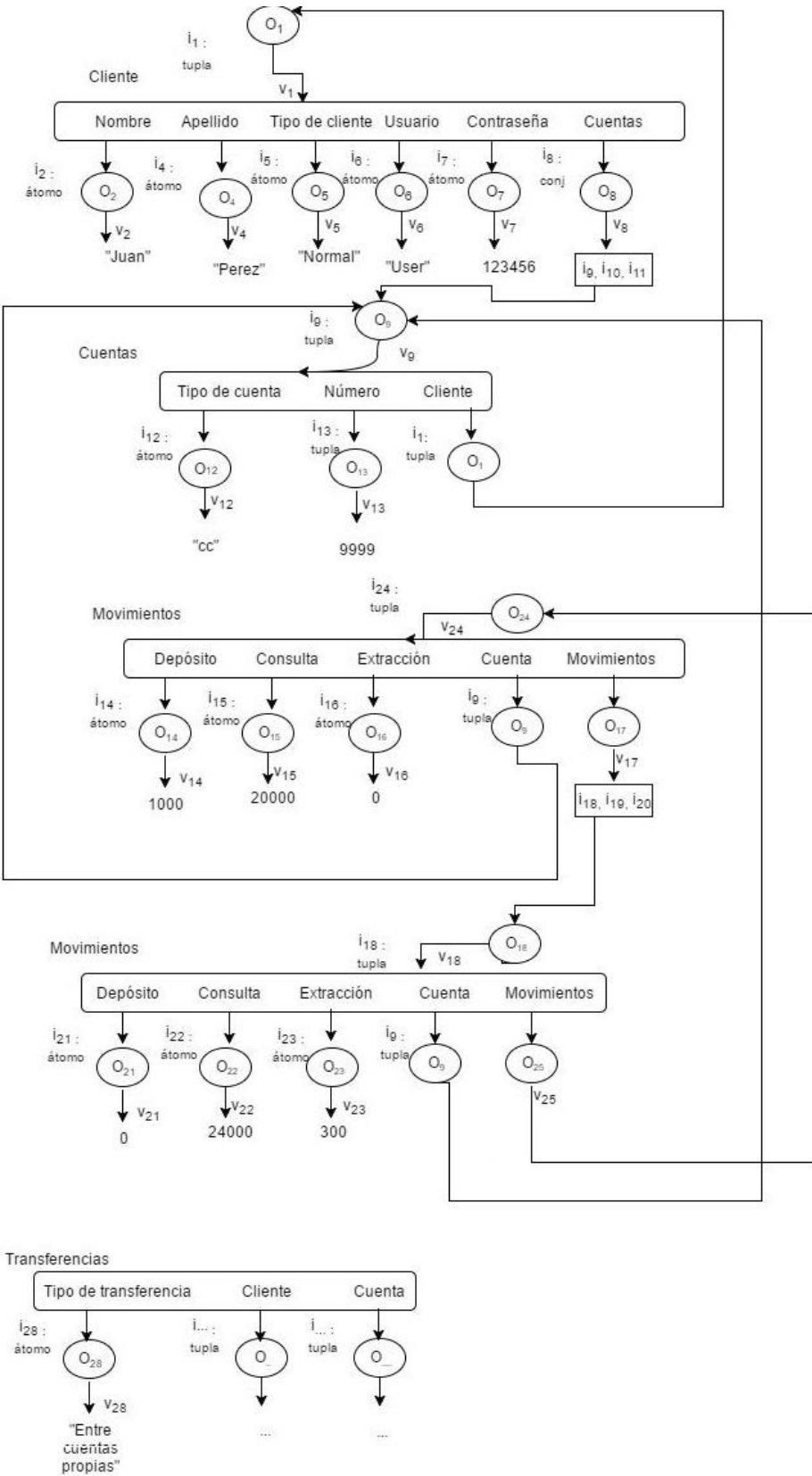


GRAFICO DE OBJETOS



DESCRIPCION LITERAL DE OBJETOS

O1 = (i1, tupla,< Nombre:i2;Apellido:i3;Tipo de cliente:i5;Usuario:i6;Contraseña:i7;Cuentas:i8>)

O2=(i2,atomo,"Juan")

O4=(i4,atomo,"Perez")

O5=(i5,atomo,"Normal")

O6=(i6,atomo,"User")

O7=(i7,atomo,"123456")

O8=(i8,conjunto,{i9,i10,i11})

O9=(i9,tupla, <Tipo de cuenta:i12;numero:i13;cliente:i1>)

O12=(i2,atomo,"cc")

O13=(i13,atomo,"9999")

O14=(i14,atomo,"1000")

O15=(i15,atomo,"20000")

O16=(i16,atomo,"0")

O17=(i17,conjunto,{i18,i19,i20})

O18=(i18,tupla ,< Deposito:i21;Consulta:i22;Extraccion:i23;Cuenta:i9,Movimientos:i25>)

O25=(i25,tupla, ,< Deposito:i21;Consulta:i22;Extraccion:i23;Cuenta:i9,Movimientos:i25>)

O28=(i28,atomo,"entre cuentas propias")

ESQUEMA ODL- DEFINICION DE OBJETOS

CLIENTE

Define Class Cliente

```
Tupla( Nombre: String;  
Apellido: String;  
tipo de cliente: String;  
usuario: String;  
contraseña: int;  
cuentas: set(Cuentas));  
operation ( crear_cliente: Cliente;  
eliminar_cliente: boolean;  
asignar_cuenta(e:Cuenta):boolean;  
quitar_cuenta(e:Cuenta):boolean;
```

End Cliente;

CUENTA

Define Class Cuenta

```
Tupla( Tipo de cuenta: String;  
Numero: int;  
cliente: tupla(cliente:Cliente);  
operation ( crear_cuenta:cuenta;  
eliminar_cuenta:boolean;  
asignar_cliente(e:Cliente):boolean;  
quitar_cliente(e:Cliente):boolean);
```

End Cuenta;

MOVIMIENTOS

Define Class Movimiento

```
Tupla( Deposito: int;
Consulta:int;
Extraccion: int;
Cuenta: int;
movimientos: tupla(movimientos:Movimientos);
cuenta:tupla(cuenta:Cuenta));
operation ( crear_movimiento:Movimiento;
            eliminar_movimiento:boolean;
            asignar_cuenta (e:Cuenta):boolean;
            quitar_cuenta(e:Cuenta):boolean;;
            crear_movimiento:(e:Movimiento):boolean;
            eliminar_movimiento:(e:Movimiento):boolean;
```

End Movimiento;

TRANSFERENCIA

Define Class Transferencia

```
Tupla( tipo de tranferencia: String;
cliente:tupla (cliente:Cliente);
cuenta: tupla (cuenta:Cuenta));
operation ( crear_tranferencia:transferencia;
            eliminar_transferencia:boolean;
            asignar_cliente(e:Cliente):boolean;
            asignar_cuenta (e:Cuenta):boolean;
            quitar_cliente(e:Cliente):boolean;
```

quitar_cuenta(e:Cuenta):boolean;

End Transferencia;

PERSISTENCIA

CLIENTE

Define ClassConjuntoClientes

Type Set (Cliente);

Operations añadir_clte(c:Cliente):boolean;

quitar_clte(c:Cliente):boolean;

crear_conj_clte:Conjunto Clientes;

borrar_conj_clte: boolean;

EndConjuntoClientes

PersistentNameTodos_Cliente:ConjuntoClientes

MOVIMIENTOS

Define ClassConjuntoMovimientos

Type Set (Movimiento);

Operationsañadir_mov(m:Movimiento):boolean;

quitar_mov(m:Movimiento):boolean;

crear_conj_mov:Conjunto Movimientos;

borrar_conj_mov: boolean;

EndConjuntoMovimientos

PersistentNameTodos_Movimientos:ConjuntoMovimientos

CUENTA

Define ClassConjuntoCuentas

Type Set (Cuenta);

Operationsañadir_cuen(c:Cuentas):boolean;

quitar_cuen(c:Cuentas):boolean;

crear_conj_cuen:Conjunto Cuentas;

borrar_conj_cuen: boolean;

EndConjuntoCuentas

PersistentNameTodos_Cuentas:ConjuntoCuentas

TRANSFERENCIA

Define ClassConjuntoTransferencia

Type Set (Transferencia);

Operationsañadir_trans(t:Transferencia):boolean;

quitar_trans(t:Transferencia):boolean;

crear_conj_transf:Conjunto Transferencias;

borrar_conj_transf: boolean;

EndConjuntoTransferencia

PersistentNameTodos_Transferencias:ConjuntoTransferencia