



Informe taller 1

Programación Avanzada

Parte 2

Integrantes:

Nombres:	Rut:	Correo:
<ul style="list-style-type: none">Diego Franco CortésRodrigo Aguilera	<ul style="list-style-type: none">21.603.362-021.760.570-9	<ul style="list-style-type: none">diego.cortes07@alumnos.ucn.clrodigo.aguilera01@alumnos.ucn.cl

Paralelo: 10796 – C1

Fecha: 14 de abril de 2024

Profesor: Tomas Alberto Reimann Beltran

Ayudantes: Edgardo Antonio Ortiz Gonzales - Marcelo Antonio Cespedes Arqueros

Requisitos

Creación y comprensión de modelos de dominio

Diagramas de clase

Implementación aplicación Java

Se nos solicita realizar el desarrollo de una aplicación capaz de procesar lectura de un archivo y almacenamiento de datos. Realizando operaciones para el correcto funcionamiento de una aplicación que será utilizada como “caja”, permitiendo la venta de videojuegos, almacenando datos sobre clientes, empleados y los propios videojuegos. Generando estadísticas según las ventas.

Puntos para considerar:

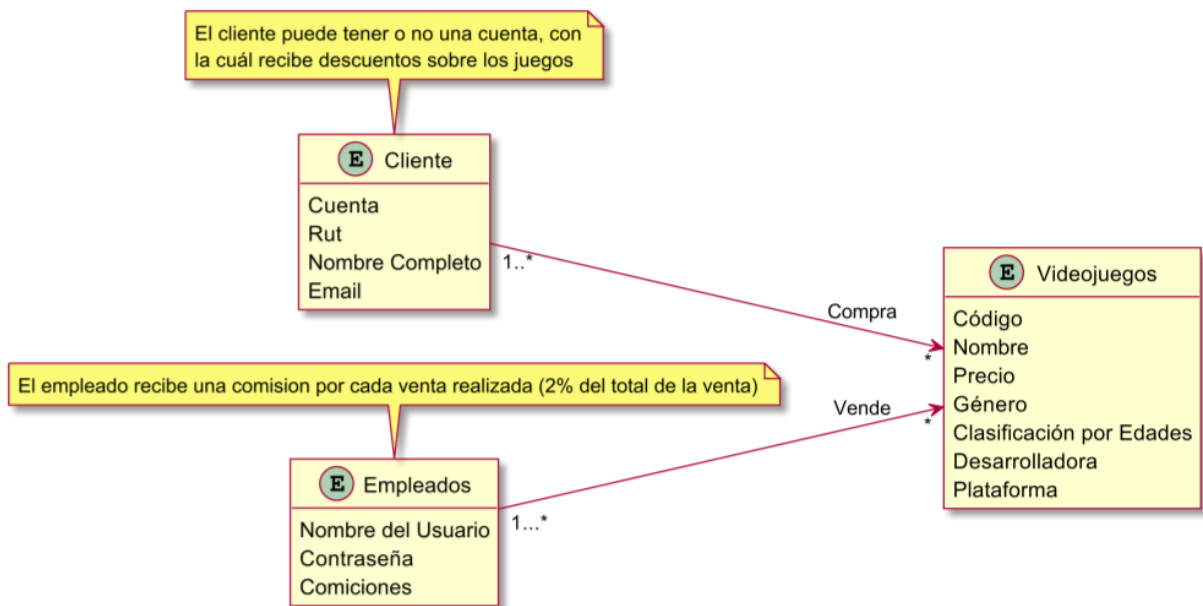
- Lectura de 2 Archivos, lista de los empleados y listado de videojuegos
- Al Inicializar la aplicación, se debe verificar la cuenta de un empleado, leído en la lectura de archivo.
- Un cliente puede poseer una cuenta de **miembro**

Se debe crear un menú donde se puedan realizar las siguientes operaciones:

- ✓ Vender un videojuego
- ✓ Buscar un videojuego
- ✓ menú estadísticas
- ✓ Cerrar la aplicación

Se considerará la creación de un modelo de dominio, diagrama de clases y posteriormente la aplicación en Java.

Modelo de dominio



Para la creación de un modelo de dominio, tomamos en consideración las 3 siguientes instancias importantes.

Empleado: como la entidad sobre la que se registrarán las ventas, considerando sus identificadores únicos como nombre de usuario, contraseña. Además, añadiendo las comisiones que obtendrá según la venta.

Cliente: como la entidad que realiza la compra, que puede o no tener una cuenta de miembro. Posee un Rut, nombre completo del cliente y un email.

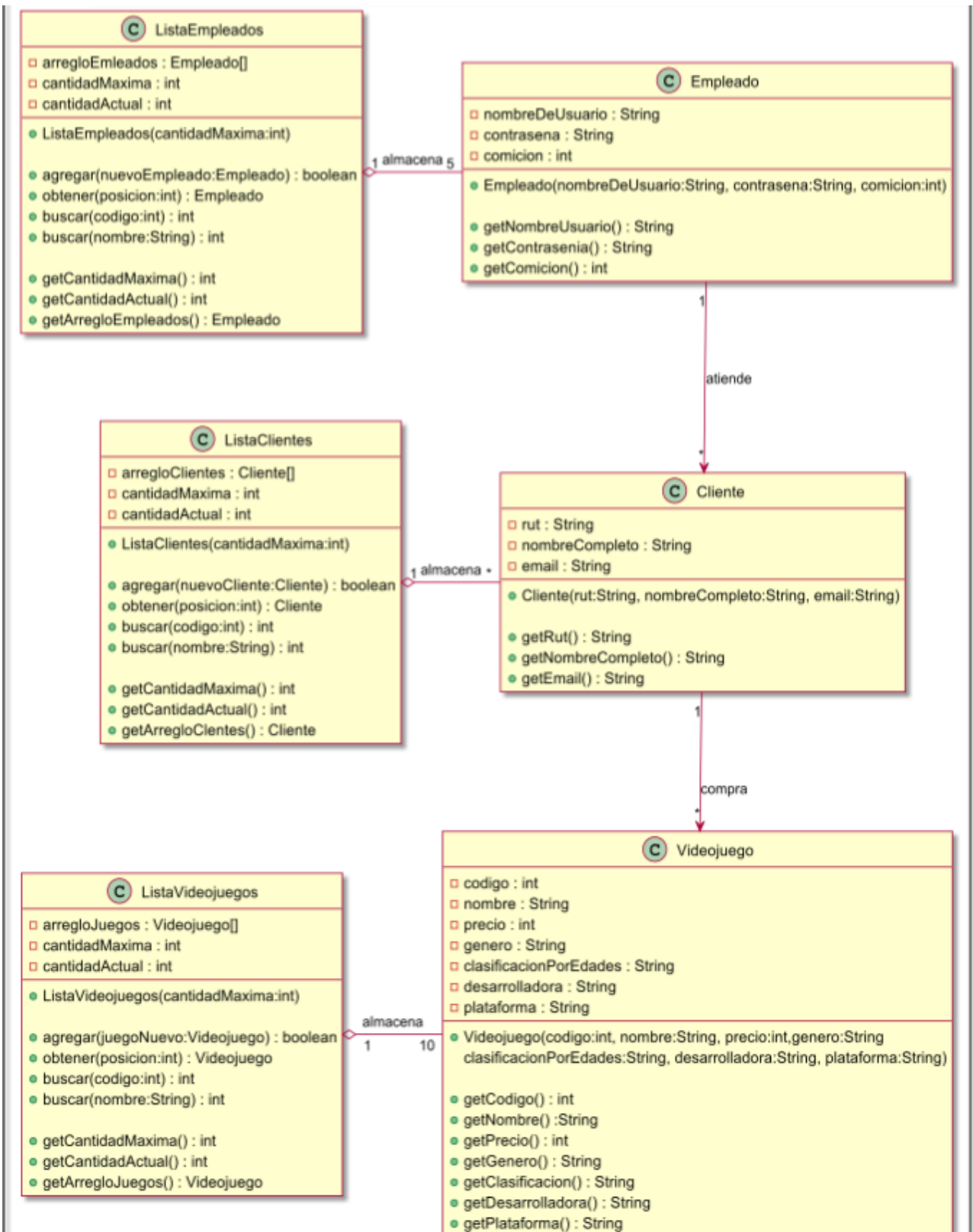
Videojuego: la entidad más importante en relación con las operaciones del programa, que se identifica con un código, nombre, precio, genero, clasificación edades, desarrolladora y plataforma.

Se relaciona de tal manera que:

Uno o muchos empleado administran muchas ventas de videojuegos, uno o muchos clientes compran muchos videojuegos.

Tiempo de dedicación taller (horas)		
Integrantes	Modelo del Dominio	Informe del taller
Rodrigo Aguilera	4 horas	1 hora
Diego Cortes	4 horas	1 hora

Diagrama de clases



Clases

Empleado

Posee nombre de usuario (String); contraseña (String) y las comisiones de venta (int).

Métodos:

En esta clase se encuentra un método constructor (crear una instancia empleado); y métodos get para obtener los datos de la instancia empleado.

ListaEmpleados

Posee un arreglo para almacenar diferentes instancias de empleados ([] Empleado); cantidad máxima de la lista (int); cantidad actual de la lista (int).

Métodos:

Método Constructor (crear el arreglo ListaEmpleados); métodos para buscar un empleado por nombre de usuario o contraseña (String); método para obtener un empleado (Empleado) ; y métodos get para obtener los datos de la lista.

Cliente

Posee rut (String); Nombre (String); Email (String)

Métodos:

Método Constructor (crear instancia de cliente); métodos get para obtener los datos de la instancia cliente.

ListaClientes

Posee un arreglo para almacenar diferentes instancias de clientes ([] Cliente); cantidad máxima de la lista (int); cantidad actual de la lista (int).

Métodos:

Método Constructor (crear el arreglo ListaCliente); método para agregar un cliente (boolean); método para buscar un cliente por rut del usuario (String); método para obtener un cliente (Cliente) ; y métodos get para obtener los datos de la lista.

Videojuego

Posee un código (int); nombre (String); precio (int); genero (String); Clasificación (String); desarrolladora (String); plataforma (String).

Métodos:

Método constructor (crear una instancia de videojuego); y métodos get para obtener los datos de la instancia videojuego.

ListaVideojuegos

Posee un arreglo para almacenar diferentes instancias de videojuegos ([] Videojuego); cantidad máxima de la lista (int); cantidad actual de la lista (int).

Métodos:

Método Constructor (crear el arreglo ListaVideojuegos); método para agregar un Videojuego (boolean); métodos para buscar un juego por código o nombre (String); método para obtener un videojuego (Videojuego) ; y métodos get para obtener los datos de la lista.

Se relaciona de tal manera que:

Un empleado se encuentra registrado en una lista de empleados, un cliente puede o **no** estar registrado en una lista de clientes y un videojuego se encuentra almacenado en una lista de videojuegos.

Uno o muchos empleados administran muchas ventas de videojuegos, uno o muchos clientes compran muchos videojuegos.

Tiempo de dedicación taller (horas)		
Integrantes	Diagrama de clase	Informe del taller
Rodrigo Aguilera	5 horas	1 hora
Diego Cortes	5 horas	1 hora