Justificación de modelo de clases

Proyecto: Red de Cuido

Curso: Diseño de Software

Profesora: Alicia Salazar Hernandez

Wilson Lopez Rubi 2014118649
Carlos Hernandez Arce 2015028168

II Semestre 2019

Tabla de contenidos

Introducción

El presente documento presenta una explicación detallada del diagrama de clases , donde se pretende explicar de manera concreta los motivos que llevaron al grupo a la solución , así mismo se realiza una explicación de los patrones de diseño que se utilizaron para el proyecto , así mismo se pretende explicar y justificar el cómo y porqué se utiliza cada patrón y que función cumple dentro de la solución.

Descripción del problema

Una empresa de red de cuido de personas, lo ha contratado para que realice un Sistema que le administrar toda su operación. La red de cuido se conforma de centros de atención en todo el país. Cada centro tiene su propio personal.

La red de cuido cuenta con diferentes niveles de servicio, pueden ser por ejemplo cuido de adulto mayor, cuido de menores, cuido por horas, cuido por días, mensuales, nocturnos, etc etc.

Se cuenta con un grupo de personas empleadas, cada una de diferentes categor□as y estudios, por ejemplo hay personas que son certificadas para el cuido de adultos mayores, otras para niños, otras tienen especialidad en personas con enfermedades terminales, etc.

Cada persona puede cumplir con una o más categorías, cada categoría tiene un precio que se le cobra al cliente, y un precio que se le paga al cuidador.

Además de los cuidadores, la empresa tiene un grupo de personas que ayudan con la parte administrativa, por ejemplo los pagos de salarios, compra de materiales, atención de llamadas, asignación de personal a solicitudes.

Cada persona tiene su puesto y salario, cada puesto tiene una lista de actividades que debe de cumplir.

Para los cuidadores, se mantiene además de todas sus características, un listado de veces que se han contratado y la calificación del servicio. Se debe mantener También un horario deseado.

Los clientes deben de registrarse primero, son evaluados para poder determinar si se aceptan o no como clients. Los clientes pueden perder su oportunidad de continuar como clientes, si un cuidador ha puesto quejas recurrentes sobre el cliente.

Se debe tener un muy buen registro del cliente, donde se puedan ver las diferentes enfermedades y tratamientos que recibe.

Una vez que son clientes, las personas tienen acceso a revisar los cuidadores y comentarios y/o evaluaciones otorgadas por otros clientes. El sitio debe de contar con características de accesibilidad, ya que el cliente puede ser la persona directa quién está contratando..

Un cliente puede hacer la solicitud del cuidador, y el Sistema debe de indicar si existe o no oportunidad de satisfacer la necesidad; si es posible, entonces se procede con el cobro del servicio por adelantado.

El administrador del lugar, debe de poder ver diferentes tipos de reportes para obtener información sobre cantidad y montos de servicios dados, por tipo de servicio, por persona, por cliente y/o fechas. Para todos los centros o solo uno.

Debe de poder verse la calificación para poder encontrar los mejores y peores cuidadores por categoría, pagos realizados vrs montos recibidos, . Para todos los centros o solo uno.

Se debe contar con al menos tres tipos de usuario, el administrador, el usuario cliente, el usuario tipo clerk.

Descripción de la solución.

Para darle solución al presente problema , el grupo de trabajo decide utilizar el modelo vista controlador como patrón de arquitectura , pues al ser un sistema que cuenta con varias sedes y la posibilidad de acceder desde diferentes partes consideramos que era necesario que la capa de datos fuera bien estructurada y que prohibiera toda la información que la aplicación requiera para trabajar.

Modelo de datos: este se implementa en sql server , pues al ser un sistema tan completo y con tanto volumen de información , una base de datos relacional puede dar mejor soporte que una que no.

Modelo de controlador : Se implementa una clase controlador en el lenguaje Java esto por su facilidad de manejar objetos , además el controlador interactuar como mediador entre todas las clases que a su vez utilizan patrones de diseño para funcionar. El controlador también tiene comunicación directa con el DBM el cual se encarga de consumir la capa de datos.

Modelo de vista: Se implementa en Java Swing , esto por la facilidad de trabajar con el mismo lenguaje que provee los demás servicios y su fácil curva de aprendizaje.

DTO como objeto de transferencia: se implementa a su vez un DTO , el cual se encarga de transportar , formatear y proveer de información al controlador está extraída desde la capa de datos o el DBM.

Diagrama de clases

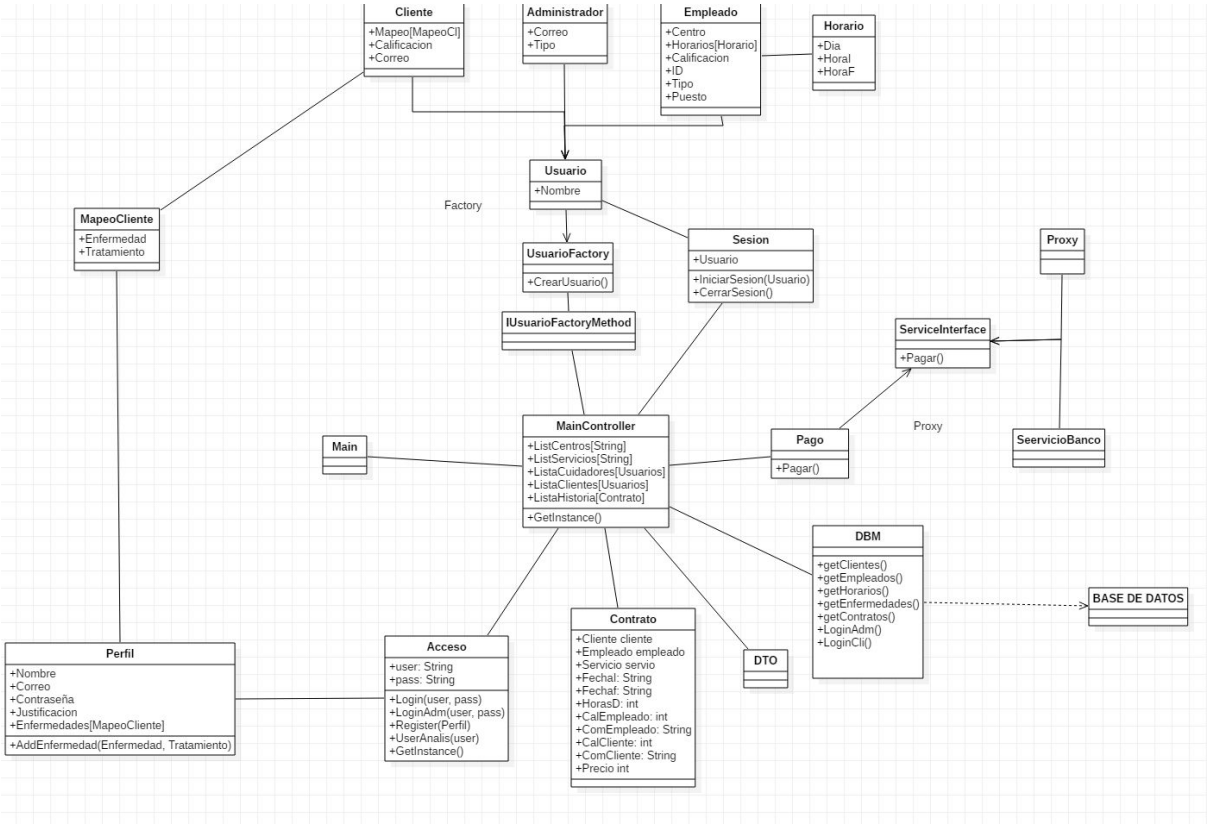
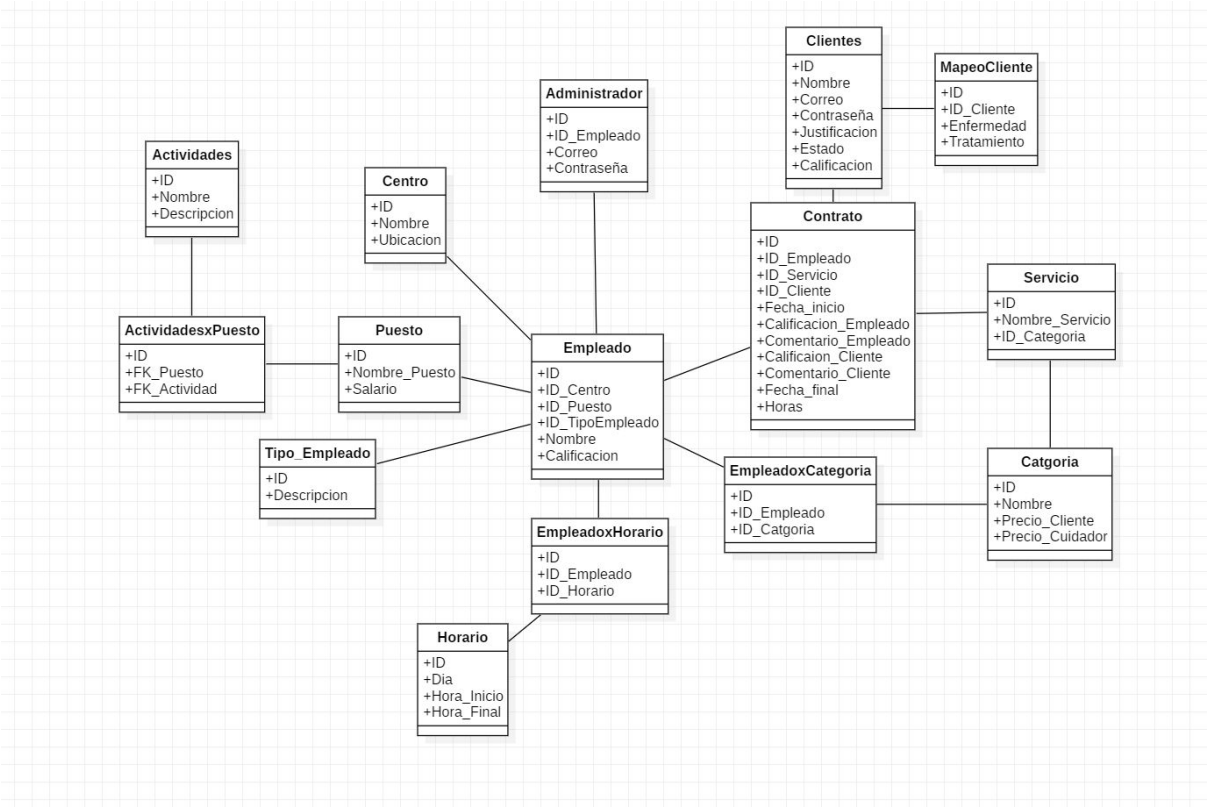


Diagrama de base de datos



Patrones utilizados

Singleton : Se utiliza en varias clases (Registró , Controlador,DBM) estas son clases que no deben ser instanciadas más de una vez y que por lo general son instanciadas varias veces y desde diferentes clases.
Factory method :Esta fábrica nos permite crear tres diferentes tipos de clientes (Administrador,Cliente,Empleado) estos son objetos que comparten algunas características , cada vez que se ocupa alguno de esos objetos la fábrica nos permite crearlo de manera sencilla invocando la clases usuario y Usuario Factory.
Proxy : Esta se implemente para proveer una conexión segura entre el sistema de pago y el banco , por motivos de alcance del proyecto no se conecta al banco realmente sino que nos da una simulación de cobro , sin embargo es importante implementar un proxy para este tipo de casos.
Object Pool : Su nombre más adecuado es pool de conexiones , pues este se implementó para proveer una interfaz entre el controlador y el DBM , permite utilizar la conexión correcta cada vez que el usuario requiere algún dato desde la base.

Justificación del diseño

Se creó la aplicación con un diseño minimalista , con colores muy neutros , para poder generar accesibilidad para las personas con discapacidad visual , así mismo por ese motivo las letras que se utilizan son con un diseño muy estándar y con tamaños adecuados para personas adultas mayores que podrían ser clientes potenciales de la app , así mismo nos esforzamos por mantener informado al usuario en todo momento , esto limita la posibilidad de errores , además esto permite que las aplicaciones de lectura de pantalla puedan guiar mejor a personas no videntes.

Por otra parte se buscó la distribución de los botones fuera lo más simple posible para que el usuario tenga claro siempre cómo hacer uso de las funciones de la misma, es por eso que se eligió que el menú fuera con pestañas a la izquierda para que los usuarios siempre el menú a mano.