

Trabajo Practico Anual Entregables

Entrega 2

Materia: Diseño de Sistemas

Curso: K3002

Grupo: 4

Integrantes:

Chiara Beretta - Guadalupe Garcia Fariñas,

Ignacio Piloni - Nicolas De Rosso

Marcos Gurruchaga - Mora Rodriguez

Profesor: Luciano Straccia

Año: 2023

Diagrama de Clases

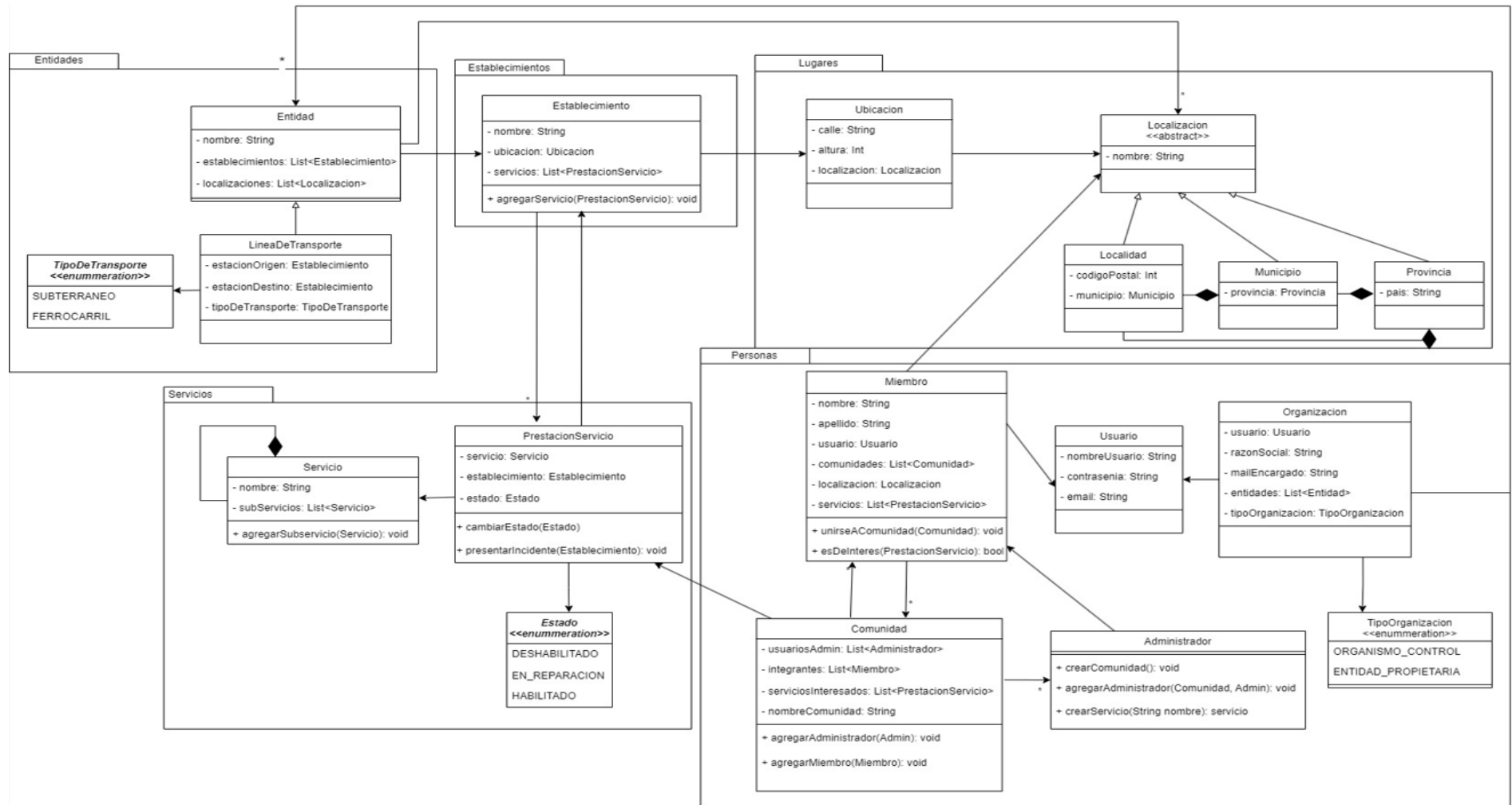
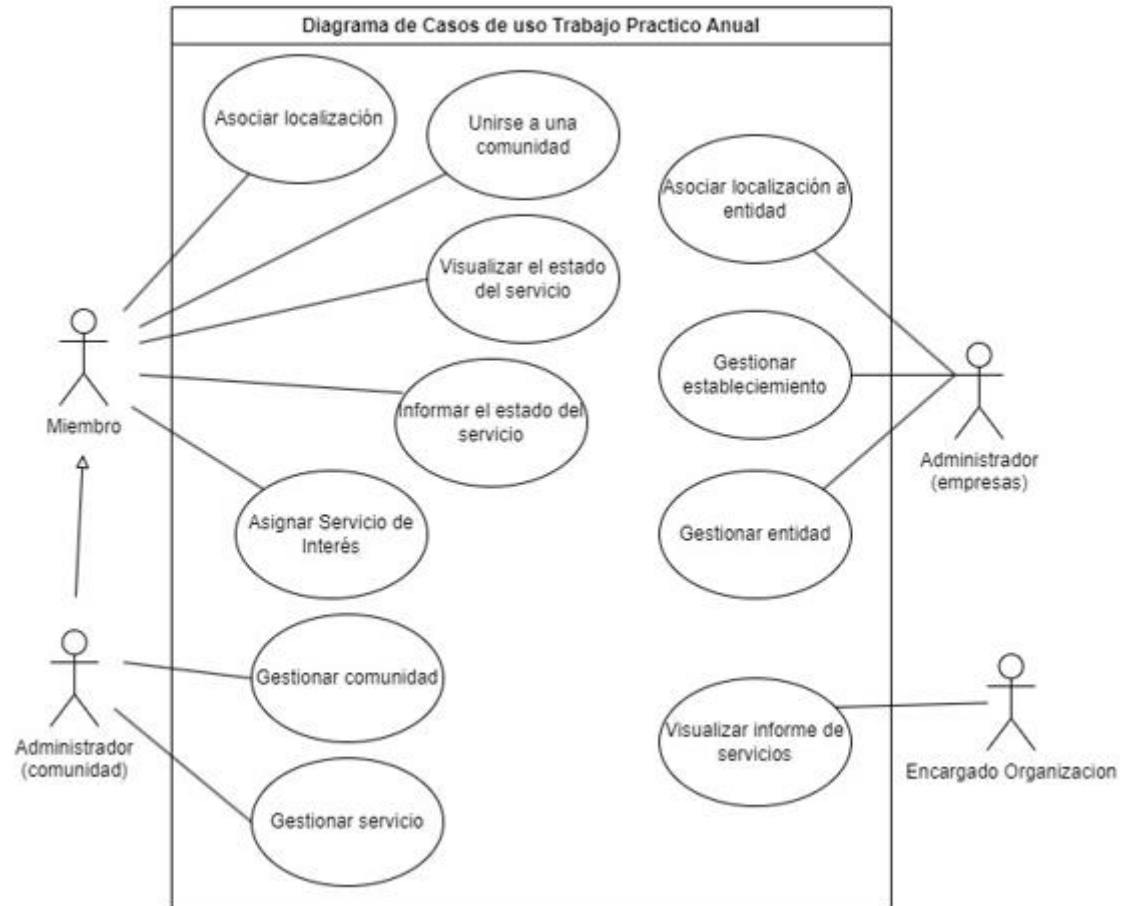


Diagrama de Caso de Uso



Decisiones de diseño

Ambito de decision (Objetos / Arquitectura / Persistencia / Otro)	Componente/s impactado/s	Decisión	Otras Alternativas	Justificación de la decisión
Objetos (Diagrama de clase)	Modelado de Servicios Públicos	Modelar una clase LineaDeTransporte, la cual contenga una lista de Estaciones	Modelar una clase Ferrocarril y una clase Subterráneo	Se trata a los ferrocarriles y subterráneos como la misma clase, lo cual permite polimorfismo y evita repetición de código.
Objetos (Diagrama de clase)	Modelado de Ubicaciones	Modelar una clase Ubicacion con sus respectivos atributos	Modelar Ubicacion como un String	- Permite reutilizar a una instancia de la clase Ubicacion para estaciones que compartan la misma ubicación real
Objetos (Caso de uso)	Modelado de actores	Modelar solo al actor "Miembro"	Modelar al actor "Miembro" y al actor "Persona"	Elegimos solo modelar al actor miembro porque hacia las mismas acciones que la persona sin comunidad y era redundante hacer un actor extra
Objetos (Diagrama de clase)	Modelado de miembro y administrador	Modelar a la clase miembro de la cual hereda administrador	Modelar a la clase persona y dentro de la comunidad que figure dentro de la lista de un administrador	Para que un administrador pueda crear comunidades y tenga los metodos para gestionarlasy, agregar nuevos admins, servicios, etc

Objetos (Diagrama de clase)	Modelado de Servicios	Modelar una clase Servicio con una lista de Subservicios (de tipo Servicio) y una clase PrestacionServicio por cada servicio que se preste en una Estacion	Modelar una clase abstracta servicio la cual hereden ServicioSimple y ServicioCompuesto, con un metodo en comun que sea estaDisponible(): Boolean	Modelar una clase abstracta nos obligaria a hacer una clase Servicio por cada Estacion que exista, y eso nos evita crear la clase PrestacionServicio; en cambio al modelar una clase Servicio la tratamos de forma abstracta para cualquier estacion, y una clase PrestacionServicio por cada servicio en una estacion en particular.
Objetos (Diagrama de clase)	Modelado de Prestacion Servicios	Agregar un enumeration estado como atributo dentro de la clase PrestacionServicio	Que el estado sea un booleano llamado estaDisponible	Este enumeration serviria para mostrar con mayor exactitud el estado del servicio, si esta cancelado temporal o permanentemente, etc
Objetos (Diagrama de clase)	Clase Localización	Se decidió crear una clase abstracta para la localización	Usar atributos nulos para el pais, departamento o municipio cuando no se necesite	Mejor encapsulamiento y coherencia en el diseño: Al crear una clase abstracta para la localización, estás creando una entidad separada y específica para representar y manejar la información relacionada con la ubicación. Esto permite encapsular la lógica y los datos relacionados en un solo lugar, lo que resulta en un diseño más coherente y fácil de mantener.
Objetos (Diagrama de clase)	Clase Entidad	Usar a la clase entidad como clase	Usar a la clase entidad como clase abstracta y crear clases que hereden de ella para las distintas organizaciones	Como por ahora las distintas organizaciones no tienen distinto comportamiento, (excepto LineaDeTransporte) lo mantenemos como clase

Objetos (Diagrama de clase)	Clase LineaDeTransporte	Usar a la clase LineaDeTransporte como herencia de Entidad	Agregar los atributos de LineaDeTransporte a Entidad y utilizar la clase Entidad para cualquier entidad especifica	Permite la reutilización de los atributos que tendran todas las Entidades; mejora la modularidad debido a que se estructura más el código; permite que sea extensible dado que cualquier Entidad que tenga atributos propios serán herencia de la clase
Objetos (Diagrama de clase)	Clase Prestacion Servicio	Mantener la localizacion en el establecimiento	Agregar la localizacion a la prestacion del servicio	La localizacion del servicio se puede saber preguntando en que establecimiento se encuentra la prestacion y no hace falta agregar el atributo dentro de cada prestacion
Objetos (Diagrama de clase)	Clase Establecimiento	Crear la clase establecimiento y que de ella se creen las instancias de estacion y sucursal	Crear la clase estacion que herede de establecimiento	Nos dimos cuenta que, por ahora, la estacion no tiene ningun comportamiento especial para la cual deba heredar de establecimiento, es decir, no hace falta diferenciarla en cuanto a atributos a la estacion de un establecimiento cualquiera
Objetos (Diagrama de clase)	Clase Organizacion	Crear una clase Organizacion que sirva tanto para empresas, como para organismos de control y entidades propietarias	Crear una clase por cada organizacion (clase empresa, clase organismoDeControl)	Al no tener (por ahora) distinto comportamiento, optamos por generalizar a Organizacion

Objetos (Diagrama de casos de uso)	Caso de uso "Visualizar informe de servicios"	El encargado ve el informe iniciando sesión en nuestro sistema.	Que le llegue por mail / whatsapp / cualquier otro medio externo a nuestro sistema	Esto facilita el acceso y la gestión de los informes para el encargado, ya que no tiene que buscar en diferentes plataformas o correos electrónicos para encontrar la información que necesita.
------------------------------------	--------------------------------------------------	-----------------------------------------------------------------	------------------------------------------------------------------------------------	-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

Implementación de la carga masiva en archivo .csv

Lo pueden encontrar en: <https://github.com/ipiloni/DDS-TPA-Grupo4/tree/main/Entrega%202>