

Decisiones de Diseño Primera Entrega TPA:

Lineas (Clase)

Modelamos el trayecto de la línea como una lista de estaciones, donde la primera estación es el primer elemento de la lista y la última estación, el último elemento. Las estaciones están en orden en la lista. Esto es porque ya lo predefinimos nosotros, los administradores en teoría no van a poder cambiar las estaciones que hay en una línea, solo el estado de la misma. Como esto no es algo que cambie frecuentemente lo dejamos así, como una lista.

Estaciones (Clase)

Las estaciones son instancias de una clase estación, ya que todas comparten atributos y métodos. Luego cada una tendrá su particularidad, como su propio nombre, ubicación, etc...

Servicios (Interfaz)

Decidimos modelar los servicios como interfaz dado que por ahora no vemos motivo para que sea una clase, ya que las clases en sí serán cada Servicio en particular, pero aún así los podemos tratar polimórficamente a todos los Servicios desde el punto de vista de la estación si es necesario.

Medio de Elevación (Interfaz)

Decidimos modelar el medio de elevación como interfaz para que un medio de elevación pueda implementar tanto las firmas de Servicio como de un Medio de Elevación. Todavía no hay nada que nos haga pensar que debemos modelarlo como una clase abstracta, capaz más adelante lo cambiaremos.

Usuarios y Comunidades:

Comunidad (Clase)

Decidimos modelar la Comunidad con una lista de miembros y otra lista de administradores. El rol de cada usuario varía dependiendo de la comunidad, por eso se detalla en la comunidad en sí el rol. Detallar esto en cada usuario en particular por ahora es innecesario, pero en algún futuro podría ser útil.

Usuario (Clase)

Modelamos al usuario como Clase para contener tanto a Administradores de Comunidades como Miembros. Ya que en el fondo son simplemente usuarios, si son administradores o miembros eso ya va a depender de la comunidad, no de ellos. Independientemente de si la comunidad a la que pertenecían dejó de existir, ellos seguirán siendo usuarios.

Tramo (Enum)

Interpretamos que cuando habla del tramo de un medio de elevación se refiere a dónde está, si da hacia la calle o hacia el andén. Por eso lo modelamos como un Enum, porque por ahora no sabemos si queremos el dato solo para mostrar al usuario o si tiene algún otro propósito.

Género (Enum)

El género del baño lo modelamos como un Enum para poder distinguir baño de hombres del baño de mujeres simplemente, y que el usuario pueda ver así si hay disponibilidad de alguno de estos en particular.

GestorDeContrasenias (Clase):

Decidimos modelarlo como una clase cuyos métodos más importantes son para ver si

un usuario tiene una clave segura y para validar si una clave es segura. Verificando si cumple con todas las condiciones que creímos necesarias para que una contraseña sea segura.