

Diagrama:

- Agregar tipos de datos en el diagrama
- Pasar a metodos estacionOrigen y estacionDestino, no hace falta que sean atributos.
- Hacer clase Servicio donde dentro tenga un atributo tipoServicio
- Eliminar interfaces vacías
- No hay concordancia entre diagrama de casos de uso y diagrama de clases
- El usuario debe tener alguna manera de registrarse en una comunidad
- Code Smells. Ante un mismo problema implementar misma solución.
- No confundir el rol de un actor y las cosas que puede hacer con los métodos de la clase de ese actor
- Agregar asteriscos en relaciones de muchos
- Eliminar validarAdmin
- Simplificar, no meter abstracciones de más
- Unificar permisos

---

Código:

- Está muy acoplado el validador con registrar usuario
- Validador de contraseña es un nombre muy genérico (Podría ser por ejemplo validar registro de sesión?)
- En vez de allMatch podría usarse map o flatMap, igual allMatch en este caso específico puede andar
- Si se manejan excepciones, en el caso de que una validacion de bien y no retorne una excepcion, no retornar nada.
- No usar PrintL, esa informacion es difícil de capturar
- Implementar tests.