

Justificaciones de diseño - Entrega 2 TPA

- Estado y sensores de heladera
- Recomendación de puntos de colocación
- Sistema de reconocimientos a los colaboradores
- Técnicos
- Carga masiva de colaboraciones

En esta entrega nos enfrentamos a desafíos varios. Uno de nuestros objetivos era lograr un diseño que permitiera gestionar un sistema de reconocimiento a los colaboradores. Para esto planteamos varias ideas, y tuvimos dudas sobre las clases que necesitábamos para representar tanto los puntos como la acción de intercambiarlos. Finalmente llegamos a una clase `BilleteraDePuntos` que guarda los puntos actuales y un historial de instancias de la clase `"Canje"`, que es un registro de los productos o servicios cambiados por una cierta cantidad de puntos. Estos últimos los pudimos agregar como colaboraciones posibles para las personas jurídicas sin alterar el diagrama.

Sin hacer muchos cambios, también pudimos sumar a los técnicos al diseño, integrados al diagrama por el medio de comunicación que comparten con los colaboradores. Estos conocen un `"Área"` que se conforma por un punto central y un radio (kms. a la redonda) que indican la zona de cobertura abarcada. Esta clase tiene un patrón adapter integrado para que, con una API, el sistema pueda determinar si un punto pertenece al área cubierta o no.

Siguiendo esta línea también implementamos un patrón adapter para integrar una API que, en base a un punto y teniendo en cuenta otros datos como densidad demográfica, los movimientos circulatorios, las heladeras que ya se encuentran en funcionamiento y otros factores que ya tiene disponibles, recomiende un lugar idóneo para una nueva heladera.

Para manejar el estado y los sensores de la heladera modificamos la clase un poco. Inicialmente pensamos en hacer los sensores con herencia, pero rápidamente la idea fue descartada porque encontramos una solución mejor con una interfaz. La heladera tiene una lista de `Sensores`, que a su vez guardan registros de estado. Además, creamos la clase `TemperaturaHeladera`, que guarda las temperaturas máxima, mínima y actual, que ahora puede setear el colaborador jurídico que la administra.

La carga masiva de colaboradores resolvimos hacerla con la biblioteca `opencsv`, que nos sirvió para leer las líneas del archivo que luego transformamos en colaboradores y colaboraciones que guardamos en una lista dentro de un objeto que utilizamos a modo de repositorio.