

Matriz de soluciones

Problema

Solución

<p>Versatilidad de la base de datos genérica: Se nos limitaba la funcionalidad que nosotros queríamos para la base de datos al usar tipos de datos genéricos.</p>	<p>Creamos una única base de datos (si era genérica íbamos a necesitar 4 distintas) estática para poder tener todas las colecciones en la misma y a su vez nos aseguramos que haya una sola copia de la misma.</p> <p>Otra ventaja que encontramos haciéndola estática fue que se nos facilitó el acceso desde cualquier parte del programa.</p>
<p>Problema con el uso de Scanners. En primer lugar abríamos y cerrábamos Scanners en cada método que los necesitaba. Esto hacía que el programa se rompa.</p>	<p>Como primer solución los declaramos como atributo de las clases que lo usan y los inicializábamos en el constructor.</p> <p>Esto también nos daba problemas ya que las clases que lo usaban son Serializables y la clase Scanner no lo es.</p> <p>Como solución final declaramos una variable estática de tipo Scanner en la clase LoginMenu y así logramos evitar los problemas.</p>
<p>Problema al listar productos del frigobar (HashMap). Al tratar de listar por segunda vez un frigobar, se rompía el programa al usar el Iterator.</p>	<p>Hardcodeamos cada clave del hashmap en cada case de un switch en el menú. Así el usuario selecciona una opción y accede al dato.</p>
<p>Problema con archivo de habitaciones. Encontramos que, al salir del programa, o bien el archivo de habitaciones no es grabado o levantado correctamente. Actualmente tenemos ese problema.</p>	<p>Solución no encontrada aún.</p>