

Pitfalls Inspection

Date: Wednesday, June 21st, 2023.

Inspector: Yeshua Chiliquinga – Jsons.

Team Inspected: Code Warriors

The program is called S_O Management and in the packages the name of the project is like HManagement this belongs to a pitfall (**uncommunicative names**) and (**inconsistent names**).

In the distributor class there is a method that shows a menu, there is a contains digits method, there is a method that saves in json defined in the same class, all this breaks with the single responsibility principle and in turn becomes a pitfall (**large class**).

In the dispatcher class there is a method that changes the name of a dispatcher object and it is like updateContactInfo, this is a pitfall (**inconsistent name**), it should be updateName or changeName.

In the distributor class there is a method that adds a product from a distributor object and it is like updateRequest, this is a pitfall (**inconsistent name**), it should be addProduct.

In the distributor class there is a method that changes a product of a distributor object and it is like changeRequest, this is a pitfall (**inconsistent name**), it should be changeProduct.

There is a default package that has a duplicate class of the model package, all this in itself is (**dead code pitfall**).

In the product item class of the model package, there is an attribute of type Date called enterDate, which breaks with the established terminology for naming attributes, therefore this is a pitfall (**inconsistent name**).

In the product item class of the model package there are, first of all, method definitions that save to a json, another that saves products to a json, another that loads products to a json this is a

(large code) pitfall, also, saveToJson is a pitfall **(duplicateCode)**, because it was already defined in distributor.

There is a report class from the model package that has nothing and does nothing, it is **(dead code pitfall)**.

In the revenueCalculator class there is a floating attribute called expenses, it breaks the terminology by being written in the plural, this is a pitfall **(inconsistent names)**.

In the revenueCalculator class, in the calculateRevenue method there is a float variable sellPercentage this breaks the terminology by being written as a verb, this is a pitfall **(inconsistent names)**.

There is a class toSellItem from the model package that has nothing and does nothing, it is a dead code pitfall, it is also a verb which breaks established terminology, this is a pitfall **(inconsistent names)**.

In the InputMenu class of the view there are 3 methods that do nothing, createUsers, sellProducts, viewSalesReport, this is **(dead code pitfall)**.

The main is in the model.