DELIVERY APP

Project report

Prepared by:

ELMER PALMA CABAÑEZ

INDEX

PROJECT GOALS

- Create an application to check the route the delivery man will take.
- Calculate this route taking into the account the stores with the most net gain depending on the products.
- Check if the delivery can be made with the available trucks.
- Limit the number of trucks in any time to max 5 of everything with a total availability of 3 of each product.
- Develop a log that can choose the output method during runtime, chaging and adding as it is needed.
- Manage the information of the store in encoded QR images.
- Update said images according to the requests of the stores.
- Check the trucks available are enough to the request.
- If there are not enough trucks inform the user.
- Develop all of this using the design patterns taught in class.

INTRODUCTION

Delivery App

The management of delivery of products for big store product providers is something that must be handled with care because if made incorrectly can cause loose of products which in turn is lost money.

In this project we took care of this problematic using some software design patterns that were the most appropiated to the necessities of our project. There were also consideration for a usable and efective interface which shows everything the user needs to use it productively.

With the use of this application it is an important objective to facilitate and speed up the process for the delivery man and the administrator of the delivery system.

DESIGN PATERNS USED

**These are the design paterns used during the development of the application.**
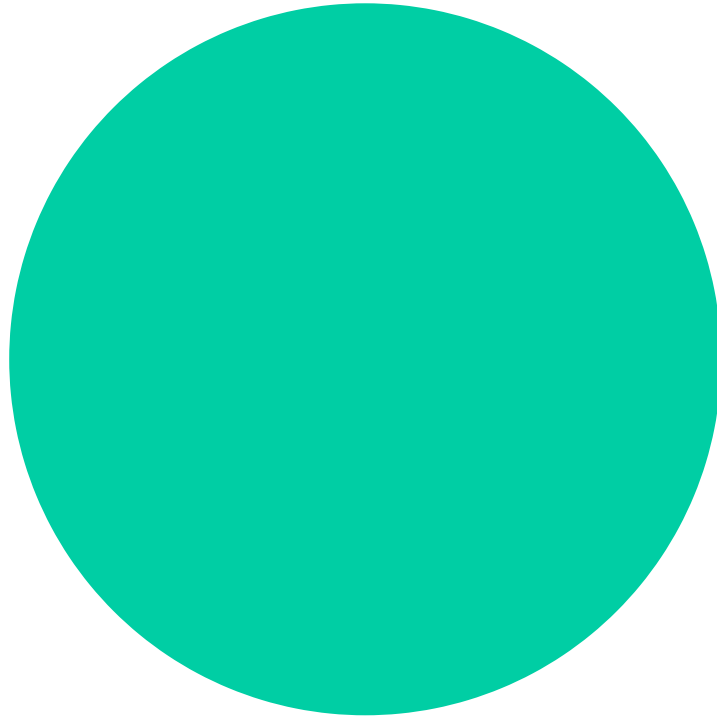
These patterns where select because they fit the project objectives of letting a more fluid a simple design for the application.

- Singleton
- Adapter
- Interpreter

SINGLETON PATTERN

This pattern allows us to create objects from a predefined set of conditions to allow us to manage and use in a more simple and comprehensible way our objects in the project. Decreasing the amount of objects created of determined class to only one, that will be used in the whole project.

It was added to allow us to manage in a more simple way the loggin of the operations done in the project, and since this is something used through the project with only one instance available at all times it was the best option to use this pattern here.
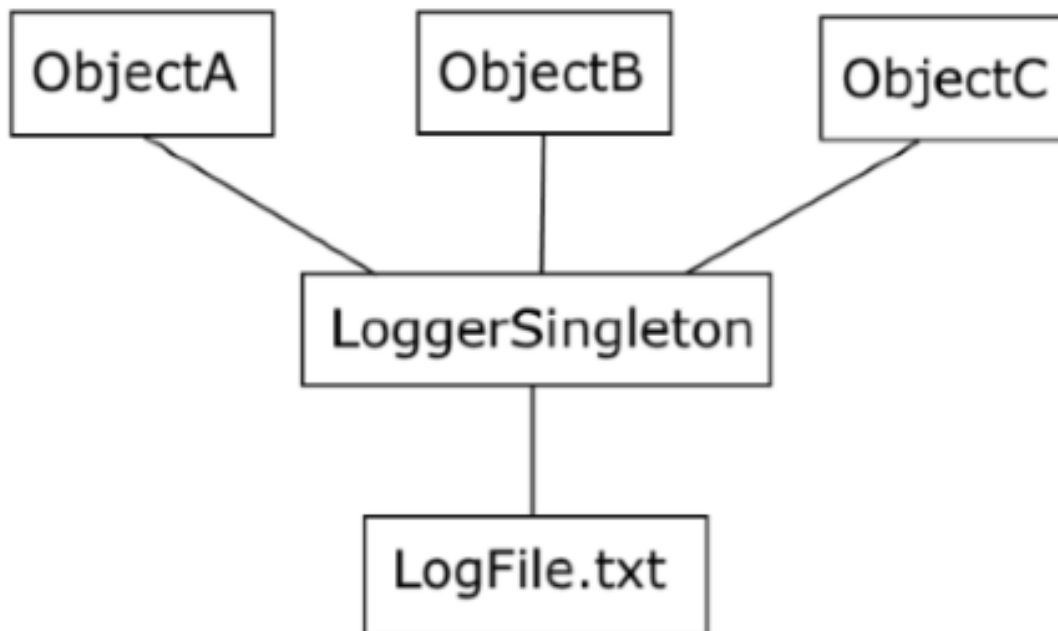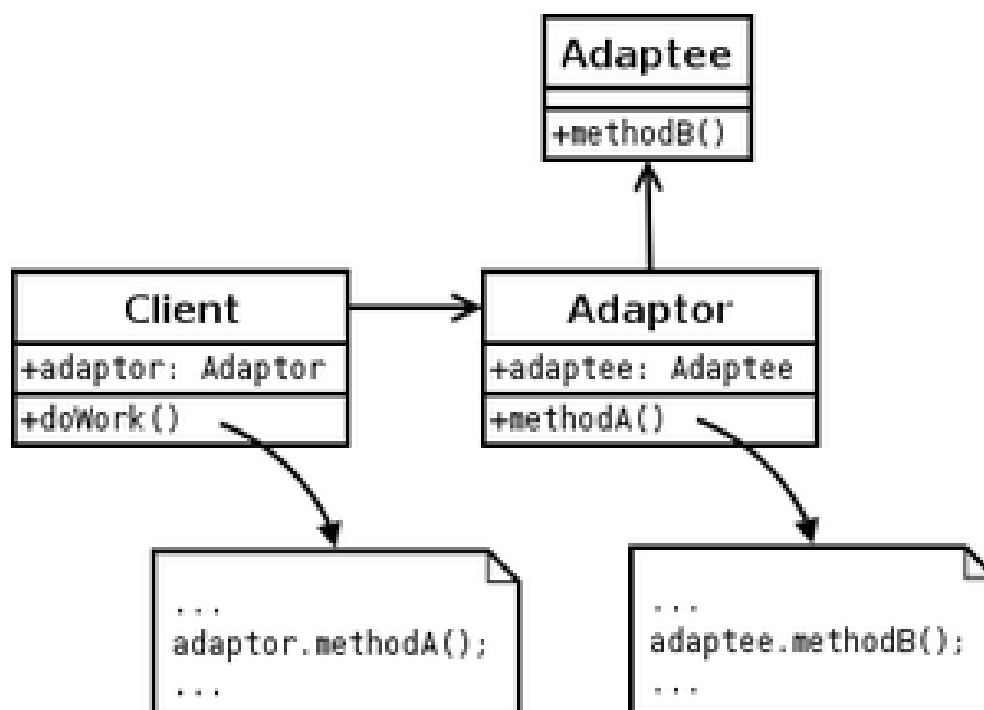
ADAPTER PATTERN

The bridge pattern was selected because it allows for a much simple managent of the system used in the project to manage, decode and encode all the qr codes pertaining to the stores and their products to be delivered.

It was added, because it allowed us to manage all the request that are related to the QR codes in just a single space and class, making debugging and managing them easier in the long run.

This pattern was used to concentrate everything about the qr management in one single space.

INTERPRETER PATTERN

The interpreter pattern allows us to manage different kinds of information in a simple manner and also allows us ot manage them in the same way independent of the information kind.

This pattern was used in the interpretation of the different kind of information needed for the differnet output options for the logger, with this we can facilitate this information to the logger independent of the output choosen.
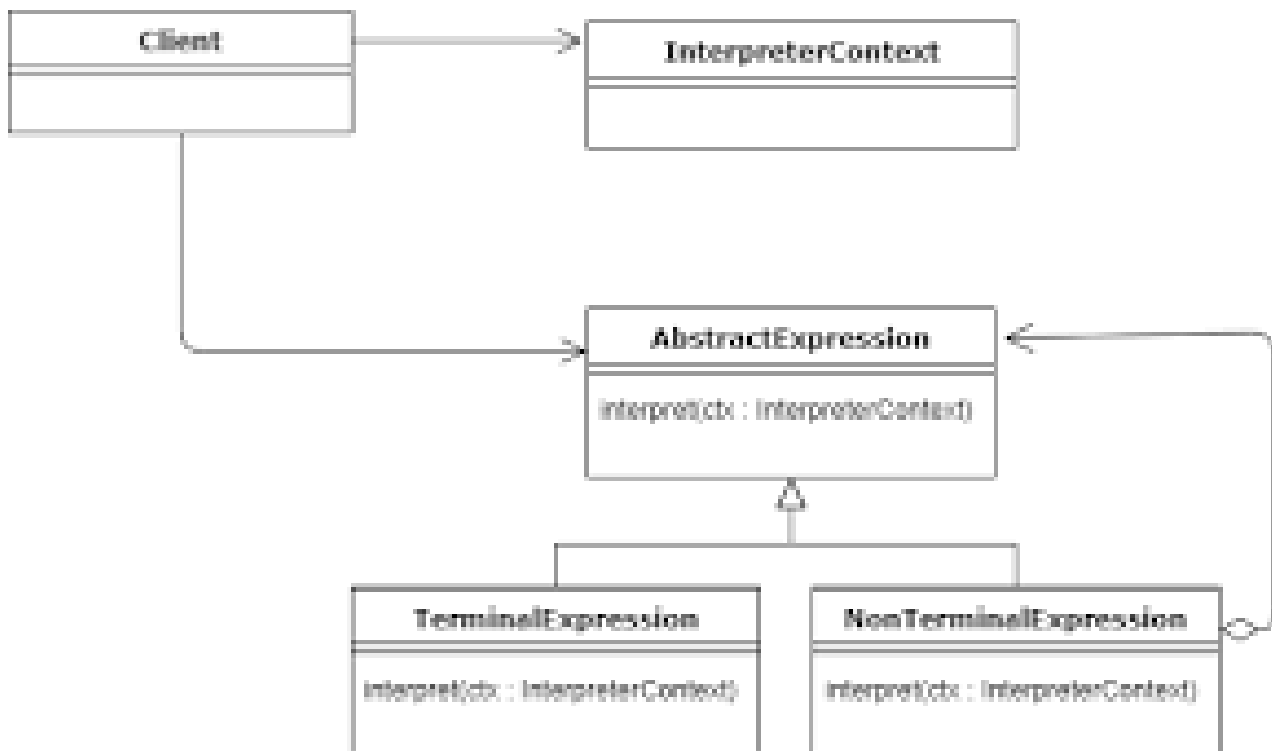
CONCLUSIONS

Final conclusions of the project development

This project was new challenge for me, because I have never programmed following any kind of design pattern, but through the development of this project I was able to see all the advantages of using these patterns, which sped development of the projecto and also left it open to be able to develop and add more functions to it, without needing a lot of time, it also would work in a proffesional environment to make my code more legible and understandable for other programmers of the team I work in. I also noticed the advantages that planned development provides for the programmer, and also the client since there are faster development times.