

# DELIVERY APP

**Project report**

**Prepared by:**

ELMER PALMA CABAÑEZ

---

# INDEX

- Front page
- Index
- Project goals
- Introduction
- Patterns used
- Singleton
- Adapter
- Chain of responsibility
- Conclusions

## PROJECT GOALS

- Create an application to access the grades system.
- For the students to be able to consult their respective grades from each of their subjects.
- Let students download a transcript of their grades.
- Let teachers consult the grades of all their respective students.
- Let supervisors, students and teachers check their profiles with all their personal information.
- The application design must use some of the design patterns seen in class, such as abstract factory, singleton, etc.
- The internal design of the application must let and be open to future additions without the necessity of a lot of extra work.
- It must have an simple to use interface.
- The application must allow the use of a configuration file to determine the the format of the downloaded transcript.
- The application will also use information obtained from a database that will be presented in this case in a DLL.

# INTRODUCTION

## Alumni 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.

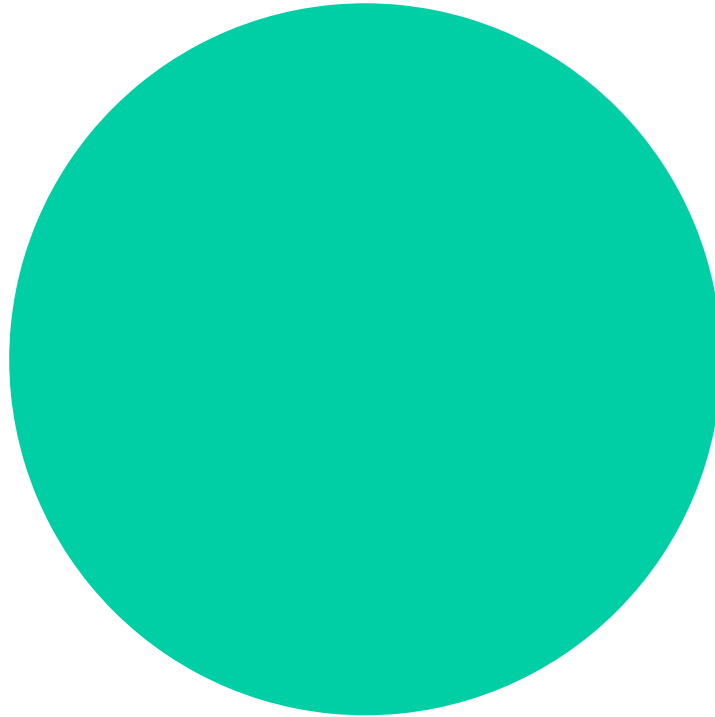
In this project we took care of this problematic using some software design patterns that were the most appropriated to the necessities of our project.

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 PATTERNS USED

These are the design patterns used during the development of the application.



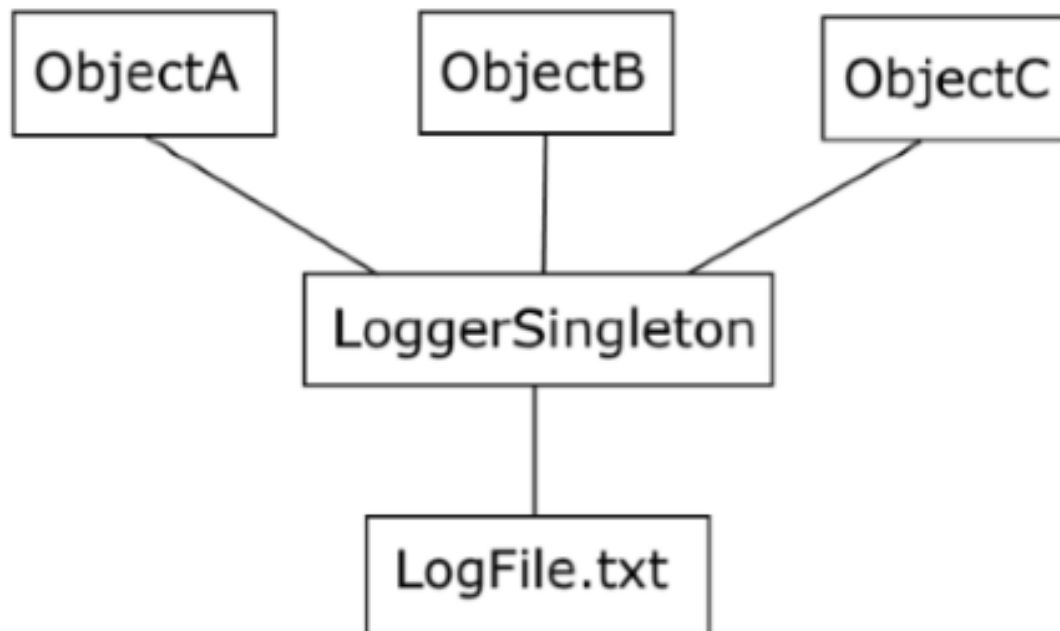
These patterns were selected because they fit the project objectives of creating a more fluid and simple design for the application.

- Singleton
- Adapter
- Chain of responsibility

## 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.

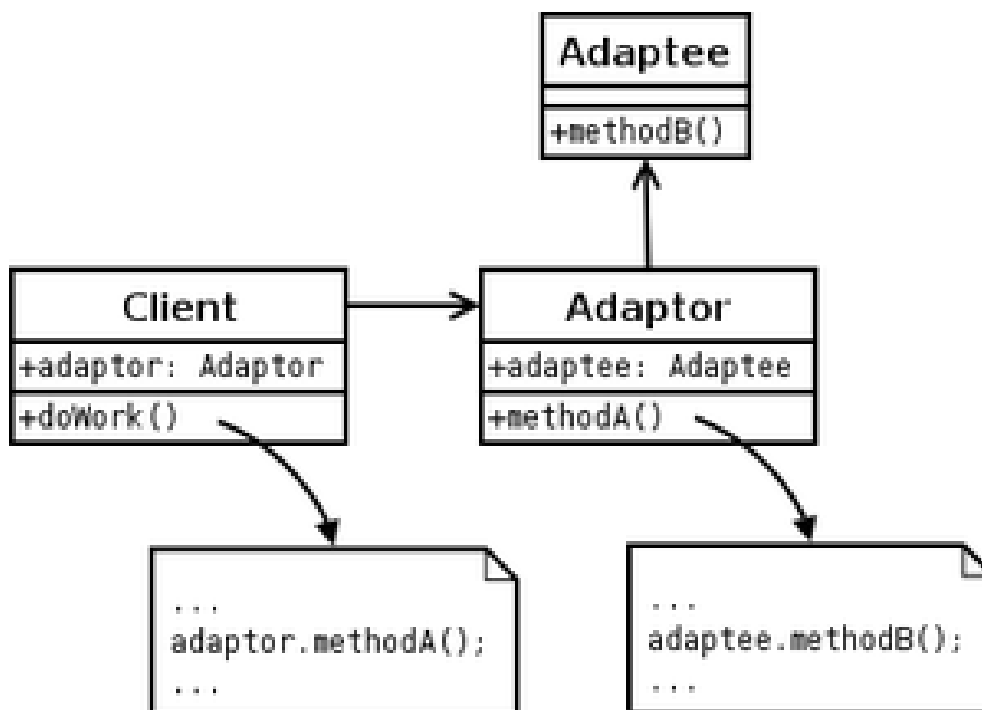


## ADAPTER PATTERN

The bridge pattern was selected because it allows for a much simple managment 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.

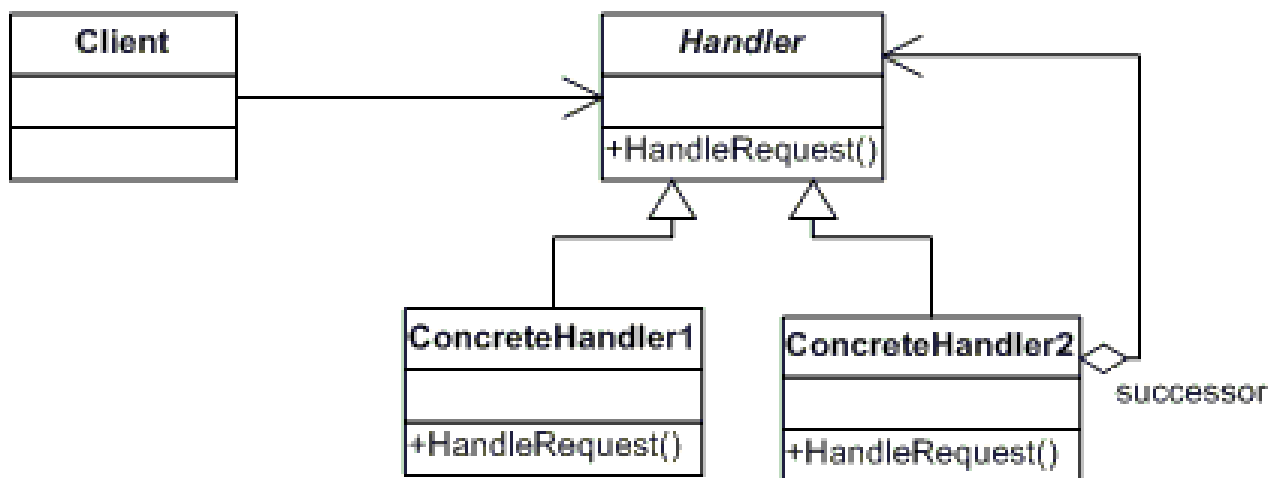
This pattern was used to concentrate everything about the qr management in one single space.



## CHAIN OF RESPONSABILITY PATTERN

The chain of responsibility pattern allows us to manage and use very complicated and long systems without many difficulties added when debugging and producing this systems.

For this reasons, this pattern was added to facilitate the development of the process of creating, reading and storing the stores information thourgh the system.





# 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 project 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 professional 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.