

OBJECT-ORIENTED SOFTWARE ENGINEERING (2EL1520)

ENGINEERING CURRICULUM: 2nd YEAR

myVelib Project

Professors:

Paolo BALLARINI
Arnault LAPITRE

paolo.ballarini@centralesupelec.fr
arnault.lapitre@centralesupelec.fr

Authors:

Osman MONLA (2101461)
Louis LHOTTE (2101405)

osman.monla@student-cs.fr
louis.lhotte@student-cs.fr

2022/2023 – 2nd Semestre

Contents

1	Introduction	2
2	Task distribution	2
3	Classes	3
4	Ride costs	3
5	Planner	3
6	Bike rent and Bike park	3
7	Statistics	3
8	JUnit tests	3

1 Introduction

This work represents the finality of the Object-Oriented Software Engineering (EL1520) course at CentraleSupélec. This project - myVelib - consists of writing an application for bike sharing. It is designed for city inhabitants to allow them to rent bikes in order to move around a metropolitan area. The appropriate way to attack this project can be deduced from a study of its complexity. This project is about making a lot of different objects interact with each other where each object has its own properties (attributes) and functionalities (methods). Thus, object-oriented programming seems to be an effective paradigm to adopt.

The complexity of a problem or any matter comes, by definition, from the high number of interacting parts and their dependencies on each other. Indeed, myVelib handles interactions between *Users*, that wish to move from one point to another in the city by renting *Bicycles* that come in two types: **mechanical** or **electrical**; *Docking stations* where the users may rent an available bicycle of the desired type from a *Parking Slot* or return one they finished using it to a *Parking Slot*. The renting, the returning and the cost operations are all handled by a *Terminal* in the *Docking Station*.

Hence, to accurately describe the objects in interaction, we modelled them using different classes, some of which are composed of one another, and some of which are used to create subclasses. The goal was to build an application with clean code that applies the principles of Object-Oriented Programming such as code reuse, inheritance, Polymorphism and the Open-Closed principle.

A thorough reading of the system requirements led us to decide what we should use as for the implementation of the methods needed to make this system works. Thus, design patterns were used where we judged important, possible and beneficial in order to facilitate an extension/update to the application.

One should note that this final project is far from being ready to be used by an End-Customer as there is no Graphical User Interface or any front-end development (no HTML, nor CSS, nor JavaScript).

2 Task distribution

In this small section, we'll have an overview of the contribution of each member of this group. Both, Osman MONLA and Louis Lhotte, worked using GitHub using Java as a programming language, and IntelliJ as an IDE. The link to the GitHub repository, created by Louis, can be found [here](#). Note that WinUX-Sudo (found in the commits of this project) and Osman Monla are the same person.

Louis LHOTTE handled the implementation of the Cost method using the **Observer Pattern** along with the Rent Bike method, the JUnit tests and the base case scenario.

Osman MONLA handled the implementation of the planning function using a **Strategy Pattern**, along with the Park Bike method, the User and Station Balances, and the Station Sorting policies using a **Strategy Pattern**.

Both, Osman MONLA and Louis LHOTTE, worked on the architecture of this final project - myVelib - thus, the UML diagram.

3 Classes

In this section, we are going to explore the classes that we implemented in order to achieve the goal of the myVelib application.

4 Ride costs

5 Planner

6 Bike rent and Bike park

7 Statistics

8 JUnit tests