Software Modeling I
Season 2024-III
Workshop No. 2 — Creational Design Patterns

`Eng. Carlos Andrés Sierra, M.Sc.`
Computer Engineering
Universidad Distrital Francisco José de Caldas

Welcome to the *second workshop* of `Software Modeling I` course, congratulations to survive to first challenge.

As you remember, you had been hired as software engineer in a `videgames arcade machines constructor company`. You delivered a *first version* for the catalog of `machines`, and it was fine. However, the **CTO** of the company thinks it is possible to get a **new catalog version** increasing `quality` and applying *better software practices*.

Thus, *company* is asking you to deliver a new version of this internal tool, improving anything you could.

Also, some requirements have been added after *first final-user interactions*. Changes are described as follows:

1. All the **Machines** must have next *attributes*: `material`, `dimensions`, `weight`, `power consumption`, `memory`, `processors`, `base_price` and `videogames`.

2. There should be a *predefined machine* called **Dance Revolution**, with additional attributes: `difficulties`, `arrow cardinalities`, and `controls_price`.

3. There should be a *predefined machine* called **Classical Arcade**, with additional behaviors: `make_vibration`, `sound_record_alert`.

4. You need to add next *pre-defined machines* options, and you must define the `appropriate attributes` they need: **Shotting Machine**, and **Racing Machine**.

5. You must define a *predefined machine* called **Virtual Reality**, with additional attributes: `glasses_type`, `glasses_resolution`, `glasses_price`.

Carlos Andrés Sierra, Computer Engineer, M.Sc. on Computer Engineering, Titular Professor at Universidad Distrital Francisco José de Caldas.

Any comment or concern related to this document could be send to Carlos A. Sierra at e-mail: *cavirguezs@udistrital.edu.co*

6. For each *predefined machine*, you must define default `base_price`, `dimensions`, `weight`, `processors`, and `memory`.

7. The client must select one of the *predefined machines*, and then, the client must select the `type of material` for the machine. In this way, a *custom machine* is created, and registered in the system with a name `provided by` the *client*.

    - If **wood** is selected, the `weight` of the machine must be *increased* in 10%, the `price` must be *decreased* in 5%, and the `power consumption` must be *increased* in 15%.
    - If **aluminium** is selected, the `weight` of the machine must be *decreased* in 5%, the `price` must be *increased* in 10%, and the `power consumption` keeps the same.
    - If **carbon fiber** is selected, the `weight` of the machine must be *decreased* in 15%, the `price` must be *increased* in 20%, and the `power consumption` must be *decreased* in 10%.

8. The client can add just videogames to the *registered machines*, and `machine price` must be *increased* by the `price` of the `videogame` added.

    - Take this into account when a *videogame* is `removed` from the machine.
    - To add videogames, in the menu just videogames in the category related to the `predefined machine` selected must be shown.

9. The videogame could be added in `standard definition` or in `high definition`, and the `price` of the `videogame` must be *increased* in 10% if it is in *high definition*.

10. All **videogames** must have *next additional attributes*: `storytelling_creator`, `graphics_creator`, `category`, `price`, and `year`.

11. You should **reduce memory** as much as possible, so check where you can *reduce* the creation of *duplicate objects* to `avoid extra memory consumption`. Be careful with *memory references*.

You must deliver a **technical report** where the *Class Diagram* of your solution is provided; here it is recommended to think in components. Also, keep all the other diagrams you think are necessary to explain your solution.

Define a `sub-diagram` por each component where *connections* with other components should be *absolutely clear*. Also, you must write about **technical concerns** and decisions you make to create the architecture you are proposing (*design patterns* used and not used, and *SOLID principles* implementation, for example).

You must *update* the **code** and provide additions to the *simple menu* to use **pre-setup machines**, show all **machines registered**, and make **searches** by: `amount of videogames`, `type of material`, and `name of videogame`, showing *machines* and them specific information independently of the material.

- Sorry, something had been missed: every `predefined machine` must have a *color* choosed by the user, and in the `searches` the users would like to *search* by `price range`, `weight range`, and `power consumption range`.

Deadline: **Saturday, Octtuber 5, 2024, 4:00 PM.**